



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

RESTAURACE POŠKOZENÝCH AUDIO SIGNÁLŮ ZALOŽENÁ NA OPTIMALIZACI S NEURONOVÝMI SÍTĚMI

RESTORATION OF CORRUPTED AUDIO SIGNALS BASED ON OPTIMIZATION WITH NEURAL NETWORKS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Václav Čermák

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Ondřej Mokrý, Ph.D.

BRNO 2025



Diplomová práce

magisterský navazující studijní program **Audio inženýrství**
specializace Zvuková produkce a nahrávání
Ústav telekomunikací

Student: Bc. Václav Čermák

ID: 221462

Ročník: 2

Akademický rok: 2024/25

NÁZEV TÉMATU:

Restaurace poškozených audio signálů založená na optimalizaci s neuronovými sítěmi

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte princip využití optimalizace v rekonstrukci audio signálů, zejména s ohledem na využití řídkosti spektra. Nastudujte princip tzv. consensus equilibrium (CE). Následně navrhnete metodu využívající principu CE pro doplňování chybějících úseků audio signálu, která bude kombinovat klasické metody založené na řídkosti a moderní přístupy využívající hlubokého učení (prostřednictvím veřejně dostupných a již naučených sítí). Otestujte a pomocí objektivních metrik vyhodnoťte kvalitu rekonstrukce v závislosti na použitých komponentách a jejich využití v rámci CE.

DOPORUČENÁ LITERATURA:

[1] BUZZARD, Gregory T., CHAN, Stanley H., SREEHARI, Suhas, BOUMAN, Charles A. Plug-and-Play Unplugged: Optimization Free Reconstruction using Consensus Equilibrium. arXiv, 2017. <https://arxiv.org/abs/1705.08983>

[2] TANAKA, Tomoro, YATABE, Kohei a OIKAWA, Yasuhiro. UPGLADE: Unplugged Plug-and-Play Audio Declipper Based on Consensus Equilibrium of DNN and Sparse Optimization. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2023. <https://ieeexplore.ieee.org/document/10095928>

Termín zadání: 10.2.2025

Termín odevzdání: 27.5.2025

Vedoucí práce: Ing. Ondřej Mokřý, Ph.D.

doc. Ing. Jiří Schimmel, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato práce se zabývá doplňováním souvislých chybějících úseků ve zvukových signálech. Dává přehled dosavadních řešení problematiky, přičemž se zaměřuje na přístupy využívající optimalizaci. V jejím rámci byla navržena a implementována metoda založená na principu *consensus equilibrium* dále označovaná jako metoda CE nebo pouze ekvilibrium, která kombinuje různé existující metody jako tzv. agenty s cílem nalézt společné kompromisní řešení, ke kterému všichni tito agenti směřují. Kvalita rekonstrukce metody CE byla porovnána s jejími jednotlivými komponentami, které se skládaly z metod využívajících spektrální řídkosti, autoregrese a hlubokého učení. Jednotlivé výstupy byly porovnány pomocí objektivních metrik a hodnocen byl i vliv parametrů ekvilibria na výslednou kvalitu rekonstrukce. V řadě případů metoda CE překonala své jednotlivé agenty a to především v momentech, kdy všichni dospěli k dobrému řešení. Pokud řešení některého z agentů nebylo kvalitní z hlediska objektivních metrik, dokázalo jej ekvilibrium potlačit pokud ostatní agenti byli sami schopni řešení nalézt. Pokud většina agentů nebyla schopná ke kvalitnímu řešení dospět, pak i metoda CE vygenerovala horší výsledek, který většinou překonal některý z jejích agentů.

KLÍČOVÁ SLOVA

audio signál, doplňování chybějících úseků signálu, mezera, consensus equilibrium, spektrální řídkost, neuronová síť, proximální operátor

ABSTRACT

This thesis deals with inpainting of continuous missing segments in audio signals. It gives an overview of previous solutions to the problem, focusing on approaches using optimization. It proposes and implements a method based on the consensus equilibrium principle, hereafter referred to as the CE method, which combines various existing methods as so-called agents in order to find a common compromise solution towards which all these agents work.

The reconstruction quality of the CE method was compared with its individual components, which consisted of methods based on spectral sparsity, autoregression and deep learning. The individual outputs were compared using objective metrics, and the influence of the equilibrium parameters on the resulting reconstruction quality was assessed. In a many cases, the CE method outperformed its individual agents and especially in moments when all generated a good solution. If any agent's solution was not of good quality in terms of objective metrics, the equilibrium was able to overcome it as long as the other agents were able to find a solution themselves. If most of the agents were unable to come to a good solution, then also the CE method generated a worse result, which was usually outperformed by one of its agents.

KEYWORDS

audio signal, inpainting, gap, consensus equilibrium, spectral sparsity, neural network, proximal operator

ČERMÁK, Václav. *Restaurace poškozených audio signálů založená na optimalizaci s neuronovými sítěmi*. Diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2025. Vedoucí práce: Ing. Ondřej Mokrý, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Bc. Václav Čermák
VUT ID autora:	221462
Typ práce:	Diplomová práce
Akademický rok:	2024/25
Téma závěrečné práce:	Restaurace poškozených audio signálů založená na optimalizaci s neuronovými sítěmi

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....
podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Ondřeji Mokrému, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

Úvod	19
1 Teoretický úvod	21
1.1 Klasické metody inpaintingu	22
1.1.1 Autoregresní metody	22
1.1.2 Metoda grafové podobnosti	23
1.2 Využití neuronových sítí v inpaintingu	24
1.2.1 Enkodér kontextu	24
1.2.2 Generativní modely (GAN)	25
1.2.3 GACELA	27
1.2.4 Diffusion Based Audio Inpainting	30
1.2.5 Deep Prior-Based Audio Inpainting (DPAI)	33
1.3 Metody založené na řídkosti spektra	34
1.4 Consensus Equilibrium	40
1.4.1 Consensus optimization	40
1.4.2 Consensus equilibrium pro inpainting	41
1.4.3 Neexpanzivní zobrazení	42
2 Návrh a implementace	43
2.1 Výběr metod a návrh agentů	43
2.1.1 Agenti neuronových sítí	44
2.1.2 Agenti spektrální řídkosti	45
2.1.3 Autoregresní agenti	46
2.2 Architektura	47
2.2.1 Poloha mezery	49
3 Testování	51
3.1 Metriky pro testování	51
3.2 Testování parametrů řídkostních metod	53
3.3 Kombinace a nastavení agentů	56
3.4 Konvergence	58
3.5 Délka mezery	62
3.6 Váhy agentů	68
3.7 Počáteční podmínky	74
3.8 Další testy	77
3.8.1 Test řídkosti a autoregrese	77
3.8.2 Test sítě GACELA	79

3.8.3	Test na různých vstupních signálech	83
3.9	Hodnocení	84
Závěr		85
Literatura		87
Seznam symbolů a zkratk		91
Seznam příloh		93
A	Prostředí a programovací jazyk	95
B	Obsah elektronické přílohy	97

Seznam obrázků

1.1	Demonstrace úlohy inpaintingu na řečovém signálu	21
1.2	Ilustrace průběhu metody grafové podobnosti	23
1.3	Enkodér kontextu	25
1.4	Kontexty sítě WGAN	26
1.5	Struktura sítě WGAN	27
1.6	Architektura sítě GACELA	28
1.7	Diskriminátory sítě GACELA	29
1.8	Architektura sítě DPAI	33
1.9	Časový průběh signálu s řídkým spektrogramem	34
1.10	Časový průběh poškozeného signálu	36
1.11	Demonstrace měkkého prahování	38
2.1	Architektura implementované metody CE	48
2.2	Agent a jeho instance	48
2.3	Demonstrace umělého poškození signálu	49
3.1	Metoda analýzy – průběh $\ell_1(x)$ v iteracích	53
3.2	Test parametrů σ a τ pro metodu analýzy	54
3.3	Rekonstrukce signálu při testování σ a τ	55
3.4	Test konvergence – SDR v iteracích	58
3.5	Test konvergence – výstupní signály agentů	59
3.6	Test konvergence – výstupní signály metody CE	60
3.7	Test konvergence – výstupní signály pro velký krok	61
3.8	Test konvergence – SDR v iteracích pro velký krok	62
3.9	Test délky mezery – výstupní signály agentů pro mezeru 300 ms	63
3.10	Test délky mezery – spektrogramy agentů pro mezeru 300 ms	65
3.11	Test délky mezery – výstupní signály metody CE	66
3.12	Test délky mezery – výstupní signály agentů pro mezeru 100 ms	67
3.13	Test vlivu vah – SDR v iteracích pro mezeru 50 ms	68
3.14	Test vlivu vah – výstupní signály metody CE pro mezeru 100 ms	69
3.15	Test vlivu vah – SDR v iteracích pro mezeru 100 ms	70
3.16	Test vlivu vah – SDR v iteracích pro mezeru 200 ms	70
3.17	Test vlivu vah – spektrogramy metody CE pro mezeru 100 ms	71
3.18	Test vlivu vah – výstupní signály pro mezeru 200 ms, 12. iterace	72
3.19	Test vlivu vah – výstupní signály metody CE pro mezeru 200 ms	73
3.20	Test počátečních podmínek – SDR v iteracích pro mezeru 50 ms	75
3.21	Test počátečních podmínek – SDR v iteracích pro mezeru 300 ms	76
3.22	Test autoregrese a řídkosti – výstupní signály pro mezeru 150 ms	78
3.23	Test autoregrese a řídkosti – spektrogramy pro mezeru 150 ms	78

3.24	Test GACELA - spektrogramy agentů pro nahrávku klavíru	80
3.25	Test GACELA - časové průběhy agentů pro nahrávku klavíru	81

Seznam tabulek

3.1	Testování parametru τ v metodě analýzy pro $\sigma = 0,1$	55
3.2	Optimální nastavení parametrů řídkosti	56
3.3	Globální nastavení klasických agentů pro první sadu testů	57
3.4	Výsledky metrik pro test konvergence	59
3.5	Výsledky metrik pro test délky mezery	64
3.6	Kombinace vah μ v testu vah agentů	68
3.7	Výsledky metrik pro test vah agentů	74
3.8	Výsledky metrik pro test počátečních podmínek	76
3.9	Výsledky metrik pro test AR+Sparse	77
3.10	Výsledky metrik pro test sítě GACELA pro nahrávku klavíru	81
3.11	Výsledky metrik pro test sítě GACELA na rockové nahrávce	82
3.12	Průměrné hodnoty metrik v testu GACELA pro 20 nahrávek klavíru	83

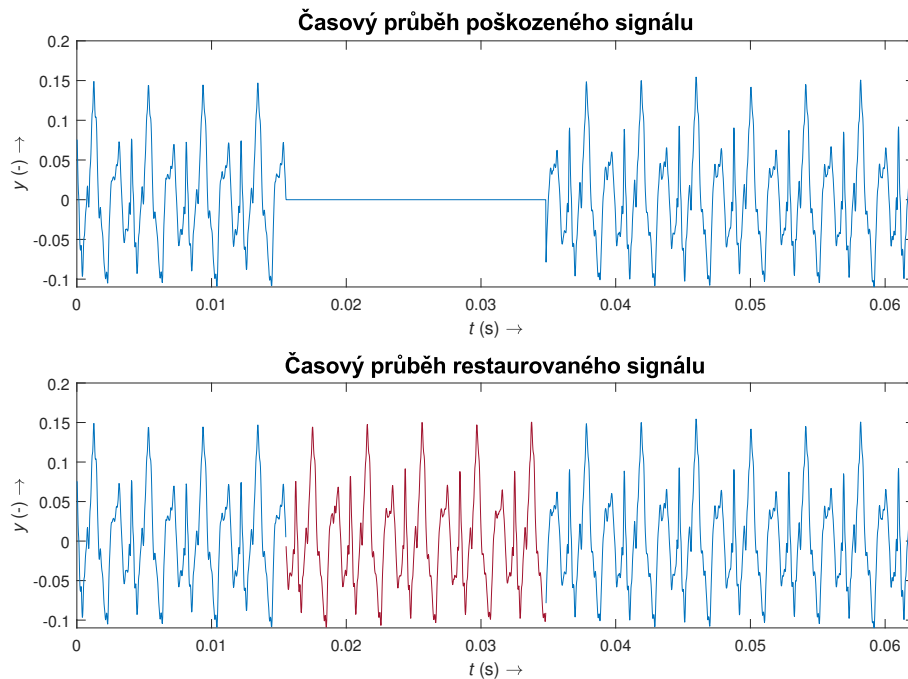
Úvod

Souvislé chybějící úseky v hudebním signálu (dále též mezery) jsou nežádoucím jevem, na který naráží různá odvětví zpracování zvukových signálů. Při záznamu nebo přenosu hudby se mohou v časové oblasti zvukového signálu objevit krátkodobé výpadky, které mohou zcela znehodnotit estetický vjem. Příčinou těchto výpadků může být například poškození hardwarových komponent v přenosové cestě, rušení s jiným elektronickým zařízením nebo chyba v nahrávacím software. K poškození zvukového signálu může dojít i později u původně neporušených záznamu například mechanickým opotřebením fyzického nosiče. Ve většině moderních postprodukčních programů pro úpravu zvuku lze tento nežádoucí jev kompenzovat manuálním střihem a vyplněním poškozeného místa vzorky z okolí (například funkce Spectral Repair z profesionálního softwaru pro úpravu zvuku iZotope RX11), což při rozsáhlejším poškození může být technicky i časově náročný proces. Tato práce se zabývá implementací metod pro automatické vyplnění těchto prázdných míst ve zvukovém signálu (dále inpainting). Jejím cílem je implementovat metodu založenou na principu tzv. *consensus equilibrium* [20] (dále CE), která vzorky v označených chybějících úsecích sama doplní. Tato metoda dokáže kombinovat různé přístupy inpaintingu, které používá jako své komponenty (dále agenty). Princip metody je obecný a lze ji proto kromě inpaintingu adaptovat na celou řadu dalších optimalizačních úloh (*denoising*, *declipping*).

V teoretické části 1 je zpracován přehled existujících přístupů pro řešení inpaintingu od prvních autoregresních algoritmů po současné metody založené na optimalizaci a neuronových sítích. U nich je sledována především popisovaná kvalita rekonstrukce poškozeného místa v závislosti na délce mezery, neboť tyto dva parametry jdou u mnoha algoritmů proti sobě a definují jeho limity. Detailněji jsou rozebrány přístupy využívající principy optimalizace, konkrétně metody řídkosti spektra, metody hlubokého učení a nakonec CE samotné. Praktická část 2 se zaměřuje na návrh metody založené na principu CE pro inpainting, její implementaci a volbu vhodných metod z teoretické části jako jejích komponent. Závěrečné testování 3 na reálných hudebních signálech zkoumá vliv parametrů CE, jejích komponent a pomocí objektivních metrik porovnává kvalitu rekonstrukce oproti klasickým metodám.

1 Teoretický úvod

Doplňování chybějících úseků zvukových signálů (dále inpainting) je jednou z optimalizačních úloh rekonstrukce poškozených zvukových signálů. Úloha má na vstupu poškozený zvukový signál, ve kterém chybí souvislé úseky vzorků. Zpravidla předpokládáme, že pozice a délky chybějících úseků jsou známy a cílem úlohy je doplnit vzorky v nich tak, aby byly zachovány původních nepoškozené části signálu (viz obrázek 1.1). Na rozdíl od ostatních úloh restaurace poškozených signálů, jako je potlačení zkreslení vzniklé překročením dynamického rozsahu (angl. declipping) a potlačení šumu (angl. denoising), je tato úloha mírně komplikovanější. Ve zmíněných úlohách je v poškozených místech stále přítomen původní signál ve své degradované podobě. Při inpaintingu vzorky v poškozeném místě zcela chybí, a je proto potřeba je vytvořit z okolního nepoškozeného kontextu. Z toho plyne vysoká míra nejednoznačnosti řešení, která roste s délkou chybějícího úseku. Různé metody inpaintingu se liší v předpokladech, které mají o vstupním signálu. Klasické metody založené na autoregresi předpokládají, že je signál v poškozeném úseku stacionární [1]. Metody využívající spektrální řídkosti tento předpoklad relaxují a zaručují pro stacionární signály pouze lepší rekonstrukci [2]. Moderní metody pro inpainting založené na strojovém učení dokáží na základě analýzy celého vstupního signálu předpokládat i složitější charakteristiky, jako je například výskyt transientu [3].



Obr. 1.1: Demonstrace úlohy inpaintingu na řečovém signálu (hláska „a“).

Tato kapitola zkoumá různé přístupy pro doplňování chybějících úseků zvukových signálů. Zmíněny jsou metody založené na autoregresi, sinusoidálním modelování a grafové podobnosti. Popsány jsou metody využívající neuronových sítí a na závěr jsou rozebrány metody založené na řídkosti spektra. Všechny metody vychází z předpokladu, že je známa poloha míst, kde došlo k poškození signálu. Samy tedy tato místa nevyhledávají, znalost jejich polohy očekávají na vstupu a soustředí se pouze na obnovení vzorků.

1.1 Klasické metody inpaintingu

Tato část slouží jako přehled klasických metod pro inpainting zvukových signálů, zejména původních modelů založených na autoregresi.

1.1.1 Autoregresní metody

Metody vychází z předpokladu, že signál v mezeře lze modelovat jako autoregresní proces. Předpokládá se, že signál je v krátkém úseku stacionární a jeho spektrální koeficienty se v tomto krátkém čase nemění. Obsah mezery je modelován pomocí vektoru autoregresních koeficientů a nepoškozených vzorků kolem tohoto úseku. S jednou z prvních metod založených na autoregresi, která se zaměřovala na zvukové signály, přišel v roce 1986 Augustus Janssen [1]. Tato metoda umožňovala spolehlivě doplnit úseky dlouhé 16 vzorků zvukového signálu při vzorkovacím kmitočtu $f_{vz} = 44,1$ kHz a až 100 vzorků dlouhé úseky řečového signálu při $f_{vz} = 8$ kHz. Nevýhodou této metody je vysoký počet autoregresních koeficientů potřebných pro získání uspokojivých výsledků s narůstající délkou mezery.

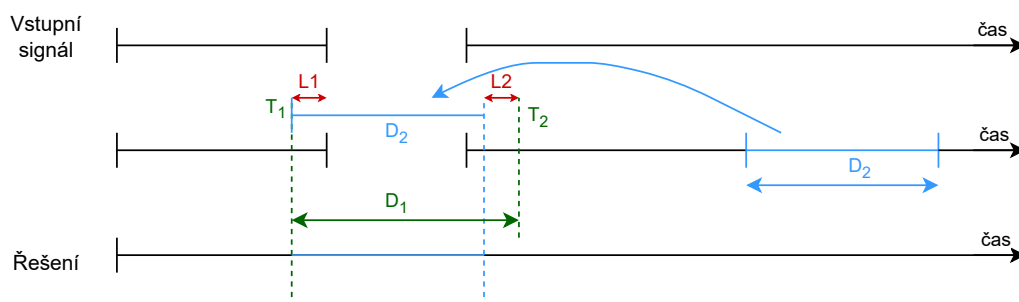
S jiným přístupem, též založeným na autoregresi, přišel v roce 1996 Walter Etter. Jeho metoda popsaná v článku [4] rozšiřuje předchozí metodu z jednoho vektoru AR koeficientů na dva pro pravostranný a levostranný kontext sousedící s mezerou. Díky tomu nemusí být součástí předpokladu o vstupním signálu stacionarita v mezeře. Etter nejprve odhaduje obsah mezery pomocí interpolační funkce a následně hledá autoregresní koeficienty odděleně pro její levý a pravý kontext. Z levého a pravého odhadu sestavuje soustavu rovnic, ze které získává signál doplněný do mezery. Metoda obchází nutnost aktualizace modelu v průběhu iterací a zaměřuje se pouze na známé úseky signálu. Neodvozují se tedy autoregresní koeficienty pro mezery, ale pro známou část signálu. Představuje několik možností interpolace pro počáteční odhad chybějících dat a zvyšuje přípustnou délku mezery až na 20 ms.

Sinusoidální modelování

Sinusoidální modelování je technika, při které je výsledný signál rozkládán na součet dílčích harmonických složek. Každá z nich je definována svým kmitočtem, amplitudou a fází. Pro inpainting lze využít metodu kombinující sinusoidální modelování a autoregresi [5]. V tomto případě jsou chybějící vzorky interpolovány pomocí autoregresního modelování amplitudy, frekvence a fáze. Autoregrese zde tedy není použita přímo k modelování vzorků, ale k modelování parametrů harmonických složek signálu, ze kterých je pak signál sestaven. Tato metoda je vhodná pro komplexní polyfonní signály a zvyšuje přípustnou délku mezery na 450 ms. Pro monofonní modulované tóny dokonce až na 1600 ms [5].

1.1.2 Metoda grafové podobnosti

Odlišným přístupem pro inpainting je využití takzvané grafové podobnosti [6]. Na rozdíl od metod využívajících autoregrese (viz sekce 1.1.1) či spektrální řídkosti (sekce 1.3) se zde při rekonstrukci nevychází pouze z okolí mezery, ale i z globálního kontextu celého zvukového signálu. Metoda o něm předpokládá řadu vlastností vycházejících z pojetí hudebních signálů jako celků složených z opakujících se částí. Napříč žánry se v instrumentaci skladeb opakují hudební témata, rytmy nebo například melodie, které autoři článku [6] označují souhrnným pojmem *redundance*. Základem metody je jejich nalezení, analýza jejich vlastností (například časových nebo spektrálních) a sestavení podobnostní mapy v rámci celého zvukového signálu (angl. *similarity graph*).



Obr. 1.2: Ilustrace průběhu metody grafové podobnosti, převzato z [6]. Segment D_2 je vybrán jako nejvhodnější kandidát pro vložení do mezery. Body T_1 a T_2 označují místa pro dosažení nejvyšší podobnosti vložení. L_1 a L_2 definují délky překrytí s původním signálem.

Doplnění vzorků do poškozeného místa pak připomíná manuální střih v postprodukčním software. V něm by byl nejprve nalezen úsek signálu s nejvyšší podobností poškozenému místu, z něj zkopírovány vzorky do tohoto místa a vyhlazeny přechody pomocí křížového prolnutí (angl. *cross-fade*). Metoda grafové podobnosti se chová obdobně s výjimkou nutnosti použít křížové prolnutí, jak ilustruje obrázek 1.2. Nejpodobnější segment je vyhledán pomocí podobnostní mapy. Pro dosažení optimálního přechodu mezi okolím chybějícího úseku a tímto segmentem jsou pomocí spektrální analýzy nalezeny body T_1 a T_2 , které označují optimální začátek a konec tohoto segmentu. Nový segment je vložen do chybějícího místa a nahrazuje okolí původního úseku o délkách L_1 a L_2 .

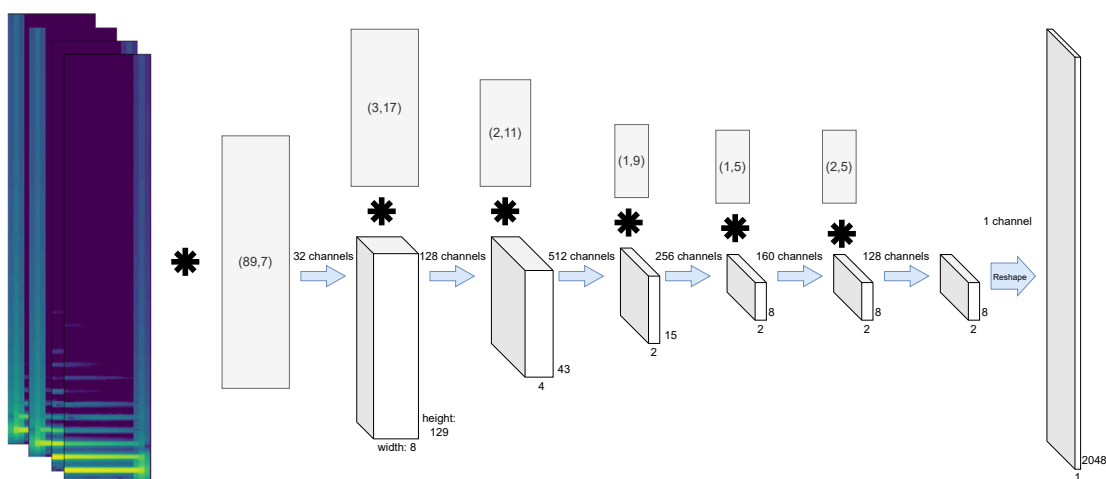
Článek [6] prezentuje dobré rekonstrukční výsledky pro délky mezer pohybující se i v jednotkách sekund. Nevýhodou metody je její omezená schopnost rekonstrukce pro žánry, které se záměrně vyhýbají opakujícím se vzorům. Největší výzvou této metody však zůstává výpočetní náročnost podobnostní mapy, která značně narůstá s délkou restaurovaného signálu i délkou mezery. Netriviální je i samotné doplnění nalezeného úseku do mezery, neboť vyžaduje nalezení optimálního místa překryvu, které nemusí být shodné se začátkem mezery. Metoda navíc oproti předchozím dovoluje změnit délku signálu.

1.2 Využití neuronových sítí v inpaintingu

Neuronové sítě zaznamenávají rozmach v mnoha různých odvětvích jako je věda, průmysl, zdravotnictví nebo například překladatelství. Díky svým schopnostem učení se složitých vztahů, rozpoznávání vzorů a zpracování velkého množství dat přináší inovativní řešení i pro restauraci poškozených signálů. Mnohé přístupy, zejména generativní modely, relaxují podmínku stacionarity signálu v mezeře.

1.2.1 Enkodér kontextu

S prvním využitím neuronových sítí pro inpainting zvukových signálů přišli autoři článku [7], který se zaměřuje na rekonstrukci hudby a hudebních nástrojů. K doplnění chybějících vzorků používá vlastní hlubokou neuronovou síť (angl. Deep Neural Network, zkráceně DNN) s plně propojenými konvolučními vrstvami natrénovanou na hudbě a hudebních signálech. Model cílí na rekonstrukci mezer o délkách v řádu desítek milisekund. Konvoluční neuronové sítě pracující se zvukovými signály přímo v časové oblasti vyžadují extrémně velké trénovací datasety k tomu, aby dosahovaly dobrých výsledků [8]. Síť v článku popsána proto kvůli redukci velikosti datasetu nepracuje přímo se signálem, ale s jeho krátkodobými spektrálními koeficienty. Skládá se ze 2 podsítí označených jako **enkodér** a **dekodér**.



Obr. 1.3: Enkodér je konvoluční neuronová síť se šesti vrstvami, převzato z [7]. Reálná a imaginární spektra levého a pravého kontextu mezery jsou zpracovávána samostatně, celkem jako 4 vrstvy. Šedé čtverce reprezentují konvoluční filtry.

Z prezentovaných výsledků v článku [7] vyplývá, že si síť ve srovnání s referenční metodou založenou na AR modelu vedla lépe pro komplexní hudební signály, ale hůře pro nahrávky samotných hudebních nástrojů. Autoři zveřejnili implementaci této sítě v jazyce Python, která je dostupná z veřejného repozitáře na GitHubu ¹.

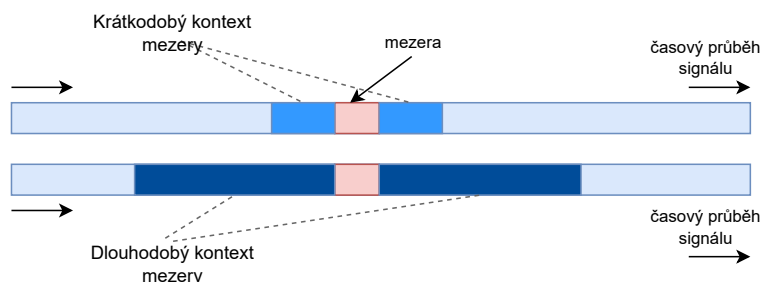
1.2.2 Generativní modely (GAN)

Kontextové enkodéry se zaměřují na doplnění mezery pomocí jejího okolí (kontextu). Generativní modely cílí na vytváření nových dat s ohledem jak na lokální, tak globální vlastnosti signálu a dovolují rozšířit přípustnou délku mezery do řádu stovek milisekund [9].

Jádrem metod založených na generativních modelech jsou dvě soupeřící neuronové sítě (angl. Generative Adversarial Networks, zkráceně GAN). Tyto sítě, **generátor** a **diskriminátor** se navzájem učí a soupeří spolu způsobem, který připomíná *hru s nulovým součtem* z teorie her. Jinými slovy, prohra jednoho je ziskem pro druhého. Generátor vytváří data, která následně analyzuje diskriminátor a určí, jestli jsou dostatečně věrná reálným datům z datasetu. Nesnaží se tedy minimalizovat nějakou vlastní regularizační funkci, ale spíše „obelstít“ diskriminátor tak, aby prohlásil jeho výstup za věrohodný [9]. Pokud diskriminátor rozhodne, že data dostatečně věrohodná nejsou, upraví generátor své parametry a vytváří nový výstup, kterým se pokouší znovu obelhat diskriminátor. Ten se naopak snaží lépe klasifikovat reálná

¹<https://github.com/andimarafioti/audioContextEncoder>

a generovaná data, čímž se síť učí navzájem. Výstupem diskriminátoru je hodnota pravděpodobnosti z rozsahu $\langle 0; 1 \rangle$, která říká, s jakou pravděpodobností jsou analyzovaná data skutečná nebo uměle vytvořena generátorem. Autoři článku [9] vytvořili model založený na GAN pro řešení inpaintingu středně dlouhých chybějících úseků (v řádu stovek milisekund), který pracuje s krátkodobým i dlouhodobým kontextem mezery (viz obrázek 1.4). Architekturu sítě ve verzi se zdvojeným diskriminátorem demonstruje obrázek 1.5.

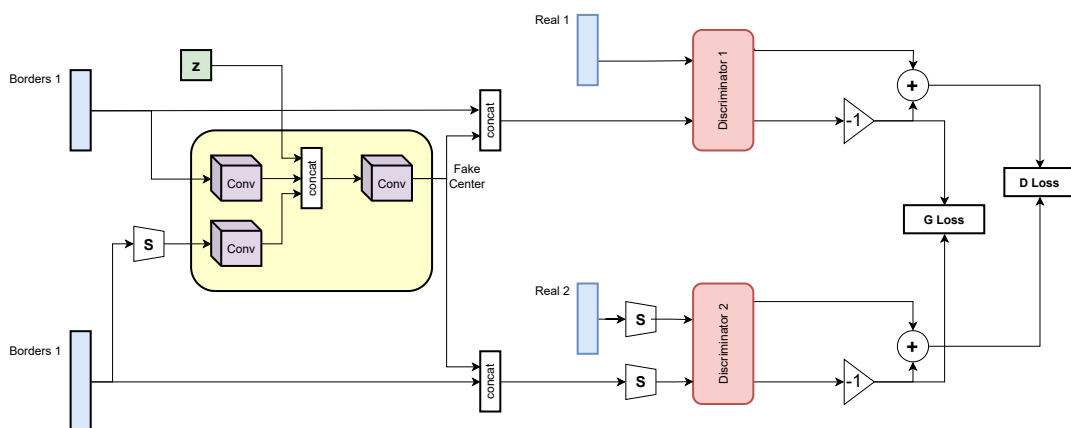


Obr. 1.4: Krátkodobý kontext je blízké okolí mezery s délkou zhruba 1 s na každou stranu. Dlouhodobý kontext je vzdálené okolí mezery, sahající 2,5 s na obě strany (převzato z [9]).

Rizikem sítí typu GAN je vznik nerovnováhy mezi generátorem a diskriminátorem. Pokud si diskriminátor bude příliš jistý ve svém rozhodování o věrohodnosti dat, bude vracet extrémní hodnoty pravděpodobnosti (0 nebo 1) a generátor tak získá minimální zpětnou vazbu pro zdokonalení svých schopností. Tento nežádoucí jev se nazývá **saturace** a je potřeba mu předcházet. Autoři článku proto nahradili diskriminátor takzvaným *kritikem*, který místo pravděpodobnosti přiděluje výstupním datům generátoru skóre (vzdálenost od reálných dat) a netrpí na defekt saturace. Použitý model se navíc snaží předejít i jinému defektu, takzvanému **kolapsu módů** [9]. Ten nastává v případě, kdy generátor najde velmi rychle výstup, kterým oklame diskriminátor a generuje pak dokola pouze tento stejný výstup. Pro inpainting středně dlouhých chybějících úseků, ve kterých může docházet k mnoha variacím v časové i frekvenční oblasti zvukového signálu, je tento jev nežádoucí, neboť možných řešení existuje mnoho a síť by generovala pouze jeden typ řešení. Autoři proto implementovali i druhou verzi modelu se zdvojeným *kritikem* pro potlačení tohoto defektu.

WGAN v obou variantách dokáže doplnit úseky dlouhé až 550 ms a dosahuje dobrých výsledků i při rekonstrukci na vysokých frekvencích. Lepších výsledků dosáhl na komplexních strukturách, které kombinovaly nahrávky více nástrojů s vyšší dynamikou a výraznými variacemi v čase. Horší výsledky pro dataset nahrávek piána byly způsobeny slabou variací hudebních signálů v datasetu, kdy dlouhodobý

kontext vzdáleného okolí mezery nepřinášel mnoho nových informací oproti krátkodobému [9]. Z těchto poznatků se dá odvodit, že síť WGAN je vhodná pro použití na komplexních hudebních nahrávkách, ve kterých se děje mnoho dynamických změn. Implementace této sítě v jazyce Python je dostupná z veřejného repozitáře na GitHubu ².



Obr. 1.5: Struktura sítě WGAN se zdvojeným kritikem (diskriminátorem) pro zabránění **kolapsu módů**, převzato z [9].

1.2.3 GACELA

Stejná skupina autorů, která vytvořila jeden z prvních modelů kontextového enkodéru [7] přišla i s inovativní generativní sítí pro audio inpainting využívající kontextový enkodér jako vnitřní součást generátoru [10].

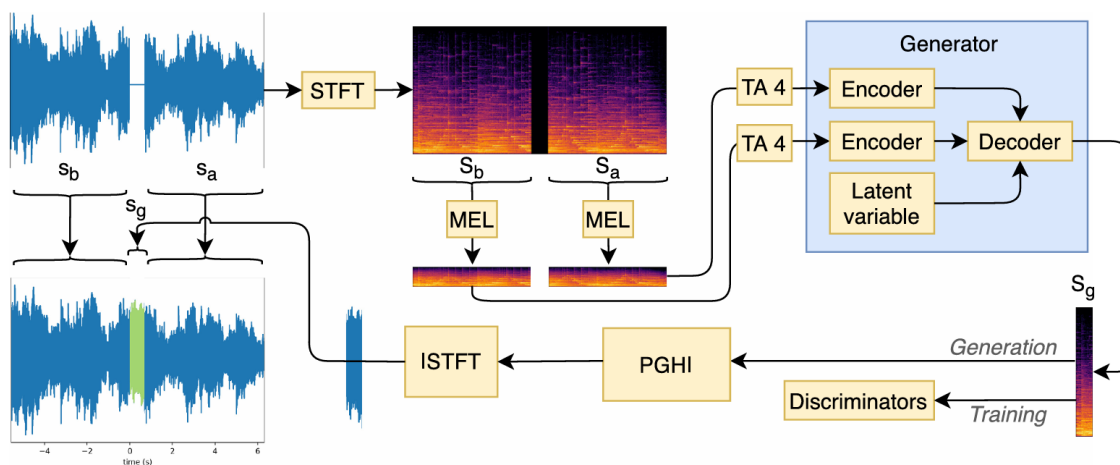
U klasických generativních sítí nemá uživatel kontrolu nad výstupem generátoru. Tuto nevýhodu kompenzují podmíněné generativní sítě (cGAN), které generátor a diskriminátor rozšiřují o další vrstvu pro zpracování dodatečných informací. Klasická síť typu GAN je závislá pouze na **latentní proměnné**, jejíž úlohou je zajistit, aby generátor produkoval různé výsledky a neomezoval se pouze na jeden výstup. Většinou je touto proměnnou šum. Vektor s dodatečnými informacemi označovaný také jako **podmíněná proměnná** je připojen k latentní proměnné a tvoří spolu vstup generátoru [12]. Latentní proměnnou může být například Gaussovský šum a podmíněnou proměnnou frekvenční složení signálu, který má být vygenerován. GACELA vychází ze sítí typu cGAN a WGAN (viz sekce 1.2.2) a cílí na rekonstrukci dlouhých mezer o délkách 375 ms až 1,5 s.

²https://github.com/nperraud/gan_audio_inpainting

Rekonstrukce takto dlouhých úseků je komplikovaná, neboť se nedá spoléhat na stationaritu signálu v mezeře a potenciálních řešení může být mnoho. Pro jeho optimální výběr zkoumá GACELA časové změny různých atributů vstupního signálu a kromě kontextu mezery využívá také **latentní proměnnou**. Cílem této sítě je vygenerovat nový obsah pro vyplnění mezery, který je konzistentní s jejím kontextem a odpovídá zvukovým charakteristikám vstupního signálu [10].

Architektura sítě

Architekturu sítě demonstruje obrázek 1.6. Vstupem generátoru jsou levý a pravý konext mezery, které jsou označeny jako s_b a s_a .

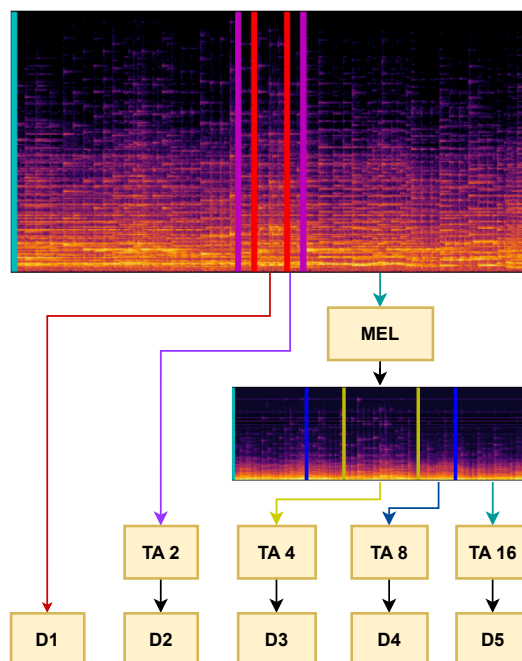


Obr. 1.6: Architektura sítě GACELA, převzato z [10].

Pomocí krátkodobé Fourierovy transformace jsou získány spektrogramy obou kontextů a následně převedeny na melovské koeficienty. Cílem těchto kroků je získat věrohodnou reprezentaci kontextu. Proto jsou v této fázi zahozeny všechny segmenty, které se nezanedbatelně překrývají s mezerou. Ze zbývajících melovských koeficientů je formováno spektrum obou kontextů S_b a S_a . Než jsou melovské spektrogramy předány generátoru, provede blok TA4 průměrování 4 po sobě jdoucích segmentů s cílem potlačit krátkodobé změny a soustředit se jen na důležité spektrální vlastnosti.

Generátor obsahuje pro oba kontexty oddělené enkodéry, jejichž melovské spektrogramy mají v generátoru roli **podmíněné proměnné**. Jejich úlohou je udržet konzistenci generovaného signálu s mezerou. Z latentní proměnné a výstupu enkodéru je sestaven spektrogram S_g , který má stejnou délku jako mezera. Síť obsahuje celkem 5 diskriminátorů, které testují sestavený spektrogram s různým stupněm přesnosti a délkou okolního kontextu. Cílem je otestovat vhodnost řešení jak v krátkodobém, tak dlouhodobém kontextu signálu, jak naznačuje obrázek 1.7.

Výsledný spektrogram S_g , který diskriminátory vyhodnotí jako věrohodný, je podroben algoritmu pro rekonstrukci fáze (PGHI, celým názvem Phase Gradient Heap Integration) [11] a nakonec pomocí ISTFT převeden do časové oblasti. Rekonstruovaný signál s_g je vložen do mezery.



Obr. 1.7: Diskriminátory sítě GACELA operující s různě širokým kontextem mezery, převzato z [10].

Hodnocení algoritmu

Autoři [10] provedli komplexní testování na čtyřech různých datasetech a kromě objektivních metrik realizovali i subjektivní poslechové testy. Jejich cílem bylo odstupňovat schopnosti sítě rekonstruovat signály v závislosti na komplexnosti vstupního signálu a délce mezery. Použité datasety byly seřazeny podle obtížnosti:

1. Zvukové signály vygenerované z jednoduchých kvantizovaných MIDI dat s malými časovými variacemi.
2. Zvukové signály vygenerované z MIDI dat nahraných muzikanty. Zde již nelze počítat s kvantizací tempa.
3. Skutečné nahrávky živých vystoupení piana. Oproti syntetickému je reálný hudební nástroj komplexnější a lze u něj vyšší variabilitu ve spektru.
4. Hudební nahrávky rockové a elektronické hudby

Testovány byly mezery o délkách v rozmezí od 375 ms do 1500 ms, přičemž délka mezery neměla významný vliv na kvalitu rekonstrukce i když sami autoři tvrdí, že jejich výsledky je potřeba podrobit bližšímu zkoumání [10]. Při poslechových testech byla většina testovaných subjektů schopna rozpoznat výskyt artefaktů, nicméně ve většině případů byly označeny za „nerušivé“ či „mírně rušivé“. Paradoxně nejhůře v subjektivních testech dopadla rekonstrukce na 1. datasetu syntetizovaného piana. I přes to je GACELA velmi slibným frameworkem pro audio inpainting dosahujícím zatím nejdelší potenciální délky rekonstruovaného úseku. Nevýhodou generativních metod může být schopnost rozeznat mezery různé délky. Pokud byl model natrénován například na mezerách o délkách 375 a 750 ms, může mít problém rozpoznat chybějící úsek o délce 450 ms [3]. Implementace této sítě v jazyce Python je též dostupná z veřejného repozitáře na GitHubu ³.

1.2.4 Diffusion Based Audio Inpainting

Difuzní modely jsou podtypem generativních modelů a využívají se též k řešení různých problémů rekonstrukce zvukových signálů, mimo jiné i pro inpainting. Jedná se o poměrně nový přístup a jeho princip spočívá ve dvou krocích:

1. **Difuzní proces**, při kterém je ke vstupnímu signálu postupně přičítán šum.
2. **Reverzní proces**, jehož cílem je „zvrátit“ difuzní proces a odstranit šum přidaný do signálu, čímž je získán rekonstruovaný signál.

Přičítaný šum zaplní i mezeru a difuzní proces z něj vytváří nová data. Jak již bylo řečeno dříve v sekci 1.2.3, generativní modely ne vždy dokážou zpracovat poškození signálu o délce, která nebyla přítomna v trénovacím datasetu. Difuzní algoritmy se snaží tento problém překonat. S prvním použitím difuzního modelu pro rekonstrukci signálu přišli autoři článku [13]. Vytvořili framework s názvem CQT-Diff založený na difuzní neuronové síti, který dokáže řešit dokonce několik rekonstrukčních úloh bez nutnosti přetrénování modelu. CQT-Diff dokáže provádět například declipping, inpainting nebo zvýšení šířky pásma.

Diskriminátor analyzuje reálná data a rekonstruovaná data generátoru, který na základě výstupu diskriminátoru upravuje svou latentní proměnnou pro dosažení lepší rekonstrukce. Difuzní modely toto párování nepotřebují a dokážou díky tomu změnit typ rekonstrukční úlohy bez nutnosti vytrénovat síť znovu. CQT-Diff pracuje v časové doméně a v rámci preprocesingu využívá konstantní-Q transformaci (angl. Constant-Q Transform, zkráceně CQT) [13] (viz sekce 1.2.4).

CQT a její inverzní forma značená ICQT jsou v síti použity pro zachycení *výškově ekvivariantní symetrie* v hudebním signálu [13]. Ekvivariance popisuje zachování vlastností při aplikaci nějaké transformace na vstup. V hudbě je ekvivariantní

³<https://github.com/andimarafioti/GACELA>

například melodie. Ta zůstane zachována i když bude celá transponována o oktávu výš. CQT je použito právě z toho důvodu, že umožňuje difuzní síti rozkrýt ekvivariance výšky ve svém spektru. Posunutí spektrálních koeficientů CQT po frekvenční ose odpovídá transpozici výšku tónu [3].

Autoři otestovali síť CQT-Diff pro inpainting na reálných nahrávkách hudby hrané na piano s dlouhými mezerami (až 1,5 sekundy). I když je testování inpaintingu popsáno pouze stručně, autoři porovnávají CQT-Diff se sítí GACELA (viz sekce 1.2.3) a tvrdí, že dosahuje podobných výsledků při větší flexibilitě použití i na jiné úlohy. Implementace této sítě v jazyce Python je dostupná z veřejného repozitáře na GitHubu ⁴.

Constant-Q transform

Konstantní-Q transformace vychází z diskretní Fourierovy transformace (dále DFT) a cílí na použití v hudebních signálech. Snahou je kompenzovat problém s rozlišením DFT na nízkých kmitočtech. Lidské ucho ale vnímá výšku tónu logaritmicky a rozlišuje takto například hudební intervaly. Kmitočtové spektrum DFT je ale rozloženo rovnoměrně s pevným krokem, což v případě hudebních signálů způsobuje poddefinovanost na nízkých a předdefinovanost na vysokých kmitočtech. CQT proto dělí frekvenční spektrum logaritmicky, čímž se snaží přizpůsobit právě vnímání lidského ucha [14]. Je definována předpisem:

$$X[k] = \sum_{n=0}^{N[k]-1} w[n, k] x[n] e^{-j\omega_k n}, \quad (1.1)$$

kde $X[k]$ je k -tý spektrální koeficient CQT, $x[n]$ je vstupní signál, $w[n, k]$ je okénková funkce pro k -tou spektrální složku, ω_k je kmitočet k -té složky a $N[k]$ definuje šířku pásma pro aktuální koeficient (převzato z [14]).

Frekvenční složky jsou rozloženy do geometrické řady. Pro hudební signály se často používá frekvenční rozložení pro temperované ladění, kdy jednotlivé složky odpovídají vzdálenosti půltónu a pro k -tou složku pak platí:

$$\omega_k = (2^{\frac{1}{12}})^k \cdot \omega_{min}, \quad (1.2)$$

Cílem je pro všechna frekvenční pásma udržet konstantní kvalitu Q. Zvolená šířka pásma závisí na zvolené okénkové funkci a v tomto případě platí:

$$Q = (2^{\frac{1}{12}}) \cong 17. \quad (1.3)$$

Různé délky okna pro každý kmitočet umožňují analyzovat přesněji. Nízké kmitočty potřebují delší časové okno, aby bylo možné zachytit celou oscilaci. Vyšší kmitočty naopak kratší rámec, aby bylo možné zachytit jejich jemnější změny v čase.

⁴<https://github.com/elaimoliner/CQTdiff>

CQT-Diff+

Autoři navazují na CQT-Diff v dalším článku [3], kde se již věnují výhradně inpaintingu. Nová metoda pojmenovaná jako CQT-Diff+ se zaměřuje pouze na středně dlouhé mezery v rozmezí 25 až 300 ms. Důvodem pro omezení je zjištěná slabá schopnost modelu CQT-Diff generovat nové události při rekonstrukci dlouhých mezer. Událostí může být změna výšky tónu, přechod na jiný akord, změna tempa nebo například výskyt transientu. Model CQT-Diff sice byl schopen doplnit i dlouhé mezery, ale obsah pak nepůsobil v jejich kontextu přirozeně [3].

CQT transformace díky logaritmickému rozložení frekvenčních složek neobsahuje stejnosměrnou složku. Pro teoretické dosažení dokonalé rekonstrukce je však potřeba ji vzít v potaz. V modelu CQT-Diff to autoři vyřešili přidáním této složky do každého segmentu [13]. To však způsobovalo artefakty na nízkých kmitočtech, což autoři odůvodňují porušením logaritmického rozdělení osy [3]. Stejnosměrná složka je proto v modelu CQT-Diff+ odfiltrována, což znemožňuje dokonalou rekonstrukci, ale v praxi to nezpůsobuje problém, neboť lidské ucho ji není schopno vnímat.

Vysoká redundance frekvenční reprezentace předchozího modelu vedla ke zvýšeným nárokům na paměť a zvýšení doby trénování sítě. Nový model se ji proto snaží optimalizovat. Model nyní nepoužívá některé předchozí funkce jako absolutní pozici frekvence, která byla přidávána na vstup sítě, neboť se ukázalo, že nemá žádný významný přínos. Dalším rozšířením je schopnost učit se globální vlastnosti celého vstupního signálu, což umožňuje nacházet podobnosti mezi různými segmenty v čase. Pro inpainting hudebních signálů toto může být přínos, neboť v hudbě se různé struktury opakují. Širší kontext proto může poskytnout relevantní informace o mezeře.

Model byl porovnán s algoritmy založenými na autoregresi a spektrální řídkosti. Pro mezery od 100 do 300 ms překonal CQT-Diff+ oba referenční algoritmy jak v objektivním testování, tak v subjektivních testech poslechu, které jsou dostupné online⁵. Implementace metody v jazyce Python je dostupná z veřejného GitHub repozitáře⁶.

⁵<http://research.spa.aalto.fi/publications/papers/jaes-diffusion-inpainting>

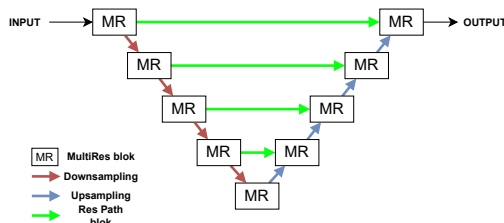
⁶<https://github.com/eloimoliner/audio-inpainting-diffusion>

1.2.5 Deep Prior-Based Audio Inpainting (DPAI)

DPAI je dalším přístupem inpaintingu založený na hlubokém učení, jehož přístup se významným způsobem liší od předchozích sítí. Využívá se zde **hluboký prior** který je použit jako regularizér. Prior lze definovat jako předpoklad o výsledných datech, které má metoda (sít) vytvořit. Oproti klasickým sítím, které jsou trénovány na velkém datasetu, tento přístup používá takzvaný *single-element training*. Místo na velkém datasetu je síť trénována na jednom poškozeném audio signálu, kterému se snaží plně přizpůsobit a zdokonalit jeho rekonstrukci. Výhodou této metody je absence nutnosti použít velké testovací datasety pro trénování, které pak může být časově náročné. Nevýhodou tohoto přístupu je trénovat síť pro každý vstupní signál znovu [15].

Algoritmus v práci implementovaný pracuje s časově-frekvenční reprezentací signálu v architektuře pojmenované *MultiResUNet*. Tato architektura využívá operací harmonické konvoluce, díky které dokáže modelovat různé charakteristiky vstupního signálu, na základě kterých pak modeluje výstupní signál.

Autoři svůj algoritmus porovnali s metodou grafové podobnosti (viz sekce 1.1.2) a algoritmem pojmenovaným CAW, kdy v rozmezí od 40 do 80 ms metoda DPAI překonala obě referenční. Obrázek 1.8 zobrazuje architekturu sítě MultiResUNet, která se skládá z konvolučních vrstev pojmenovaných *MultiRes blocks* (zkráceně MR). Tyto bloky jsou uloženy v enkodéru a dekodéru a pracují s různými rozlišeními daného signálu.



Obr. 1.8: Architektura sítě DPAI, převzato z [15].

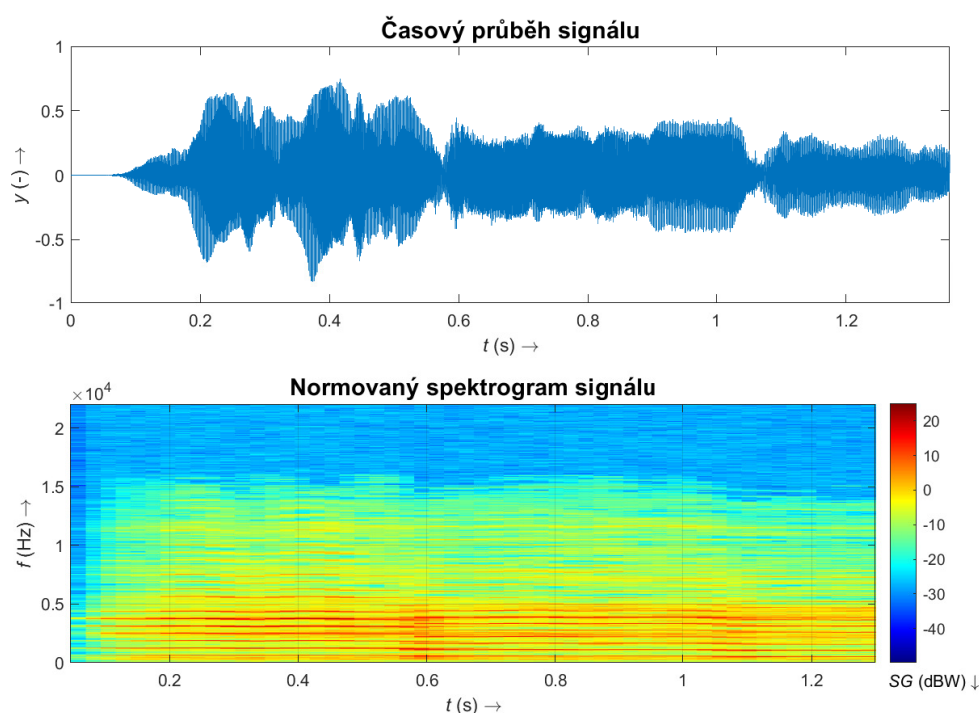
Metoda dosahuje dobrých výsledků pro krátké mezery, kde překonává referenční metody. Pro delší mezery však kvalita rekonstrukce rychle klesá [15]. Implementace metody v jazyce Python je dostupná z veřejného GitHub repozitáře ⁷ a dostupné jsou i výsledky poslechových testů⁸.

⁷<https://github.com/fmiotello/dpai>

⁸<https://fmiotello.github.io/dpai/>

1.3 Metody založené na řídkosti spektra

Dalším přístupem pro inpainting jsou metody založené na řídkosti spektra, které definují inpainting jako inverzní *ill-posed* problém. Řešení takového problému je nejednoznačné a komplikované. Vychází se zde z předpokladu, že nepoškozený signál po převodu do frekvenční oblasti vykazuje řídkost spektrálních koeficientů, tedy že jejich velká část je rovna nule (viz obrázek 1.9). I zde se takový předpoklad nazývá **prior**. Naopak signály, které mají ve svém časovém průběhu chybějící úseky, spektrum v těchto místech řídké nemají (viz obrázek 1.10). Ve spektrogramu poškozeného signálu je v místě mezery stále patrná harmonická struktura, ale koeficienty jsou méně řídké.



Obr. 1.9: Příklad časového průběhu a spektrogramu houslí hrajících legato. V záznamu je několik výrazných harmonických složek a mnoho spektrálních koeficientů blížících se nule.

Cílem metod inpaintingu využívajících řídkosti je postupné získání signálu, jehož koeficienty ve spektru jsou řídké. Způsobů, jak toho lze dosáhnout je více, nicméně obecná formulace problému je pro ně společná. S její první podobou pro použití na zvukových signálech přišel Amir Adler a jeho tým [8], avšak kvůli přehlednosti bude v této práci použita obdobná formulace převzatá z [2].

Audio inpainting založený na řídkosti lze formulovat jako optimalizační problém, jehož cílem je nalézt takový signál $\mathbf{z} \in \mathbb{C}^L$, který je řídký a mimo chybějící úseky odpovídá původním nepoškozeným vzorkům restaurovaného signálu $\mathbf{y} \in \mathbb{C}^L$:

$$\arg \min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{vzhledem k } S\mathbf{x} \in \Gamma. \quad (1.4)$$

Rovnice je formálním zápisem problému a vystupují v ní tyto proměnné, množiny a operátory:

S Operátor syntézy $S : \mathbb{C}^N \rightarrow \mathbb{C}^L$, který z N koeficientů generuje signál délky L . K němu přidruženým je operátor analýzy $A : \mathbb{C}^L \rightarrow \mathbb{C}^N$, který naopak ze signálu vytváří koeficienty. Příkladem takového operátoru A může být krátkodobá Fourierova transformace, která signál z časové oblasti transformuje do oblasti časově frekvenční a pro různá časová okna poskytuje spektrální koeficienty. Přidruženým operátorem S je pak inverzní krátkodobá Fourierova transformace, která naopak z koeficientů sestavuje zpět signál v časové oblasti. U operátorů je výhodné omezit se pouze na práci s těsnými framy (Parseval *tight frames*), které zajišťují perfektní rekonstrukci signálu a nemění jeho ℓ_2 normu při přechodu mezi prostory, formálně:

$$SA = Id, \quad \|A\mathbf{y}\|_2 = \|\mathbf{y}\|_2. \quad (1.5)$$

$\mathbf{x}$ Vektor koeficientů, který je proměnnou v minimalizační úloze. V případě STFT se jedná o spektrální koeficienty.

$\|\mathbf{x}\|_0$ ℓ_0 norma, která vyjadřuje počet nenulových prvků argumentu a definuje tak požadavek na řídkost spektrálních koeficientů. Hledat řešení optimalizačního problému obsahujícího ℓ_0 normu je ale výpočetně náročné [2]. V praxi se proto místo ní používá ℓ_1 norma a místo skutečného řešení se hledá jeho aproximace, která se k němu za stanovených podmínek co nejvíce blíží.

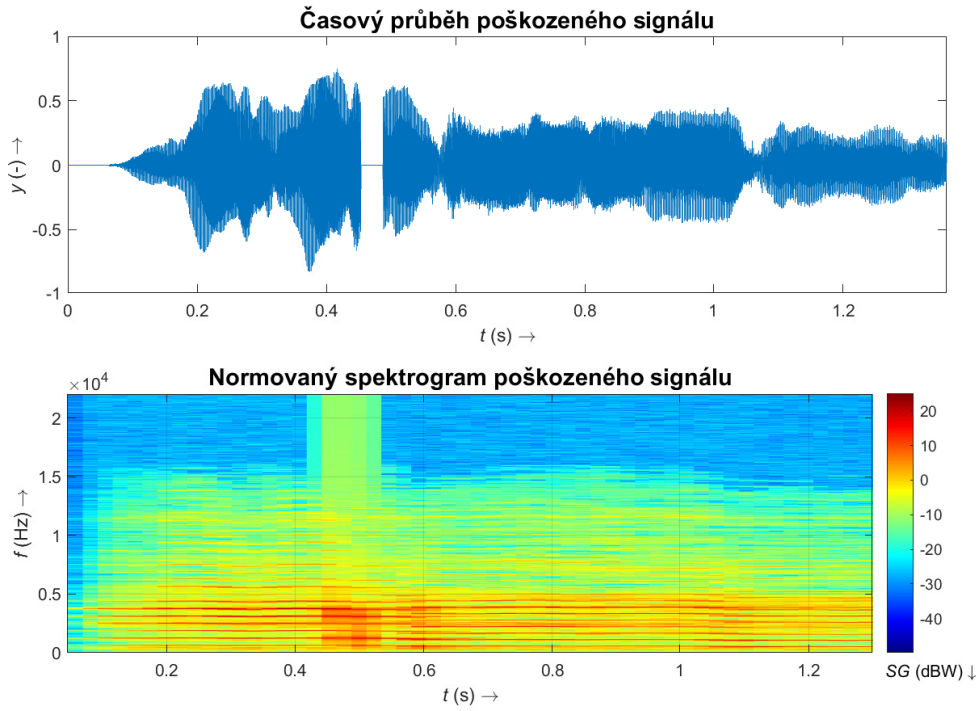
Γ Konvexní množina všech přípustných řešení, která zajišťuje konzistenci restaurovaného signálu s původními nepoškozenými vzorky, formálně:

$$\Gamma = \{\mathbf{z} \in \mathbb{C}^L \mid M_R \mathbf{z} = M_R \mathbf{y}\}, \quad (1.6)$$

kde $\mathbf{y}$ je původní signál, $\mathbf{z}$ nějaké řešení optimalizačního problému a $M_R : \mathbb{C}^L \rightarrow \mathbb{C}^L$ je operátor nazývaný též jako „reliable mask“. Tato bitová maska obsahuje nuly v místech, kde je v původním signálu poškozené místo a jedničky na zbylých spolehlivých vzorcích (angl. *reliable samples*).

Všechny tyto komponenty vymezují požadavky na hledané řešení. V kapitole 2 bude věnována pozornost vztahu Gaborovy transformace a těsných framů, které disponují některými výhodami pro použití v inpaintingu. Aproximace řešení problému se dále dělí na 2 přístupy **syntetizující** a **analyzující**, které se liší podle způsobu, jakým hledají řešení. Výše uvedená definice (1.4) je pro syntetizující model. Pro analyzující platí obdobný vztah:

$$\arg \min_{\mathbf{z}} \|\mathbf{A}\mathbf{z}\|_0 \quad \text{vzhledem k } \mathbf{z} \in \Gamma. \quad (1.7)$$



Obr. 1.10: Poškozený předchozí průběh z obrázku 1.9 nyní s chybějícími vzorky. V místech, kde došlo k překryvu s mezerou, nejsou spektrální koeficienty řidké.

Proximální operátory a inpainting

Před definicí analyzující a syntetizující metody je potřeba definovat některé pojmy, které jsou klíčové pro nalezení řešení minimalizačních problémů. Mnoho minimalizačních problémů z oblasti zpracování signálů, pod které spadá i inpainting založený na spektrální řídkosti, může být formulováno jako problém konvexní optimalizace. Jejich cílem je minimalizovat účelovou funkci, která je vyjádřena jako součet konvexních funkcí definovaných na reálném prostoru, formálně:

$$\underset{\mathbf{x} \in \mathbb{R}^N}{\text{minimize}} \quad f_1(\mathbf{x}) + \dots + f_m(\mathbf{x}), \quad (1.8)$$

kde $\forall i \in \{1, \dots, m\}$ platí, že $f_i(\mathbf{x})$ jsou konvexní funkce zobrazující z prostoru $\mathbb{R}^N$ do prostoru $\mathbb{R}$ [17]. Díky tomu, že jsou funkce konvexní, lze řešení minimalizace *rozdělit* na minimalizaci jednotlivých funkcí f_i . Tento přístup je méně výpočetně náročný, než minimalizace složené funkce a nazývá se metoda *proximálního dělení* [2]. Pro nalezení hodnoty, která minimalizuje konvexní funkci f_i slouží takzvaný **proximální operátor**, který je definován jako zobrazení $\text{prox}_f : \mathbb{R}^N \rightarrow \mathbb{R}^N$ dané pro všechna $\mathbf{x} \in \mathbb{R}^N$ a platí:

$$\text{prox}_f(\mathbf{x}) = \arg \min_{\mathbf{y} \in \mathbb{R}^N} f(\mathbf{y}) + \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|^2. \quad (1.9)$$

Pokud je funkce $f(\mathbf{y})$ indikátorovou funkcí konvexní množiny $C \subset \mathbb{R}^N$, pro kterou platí

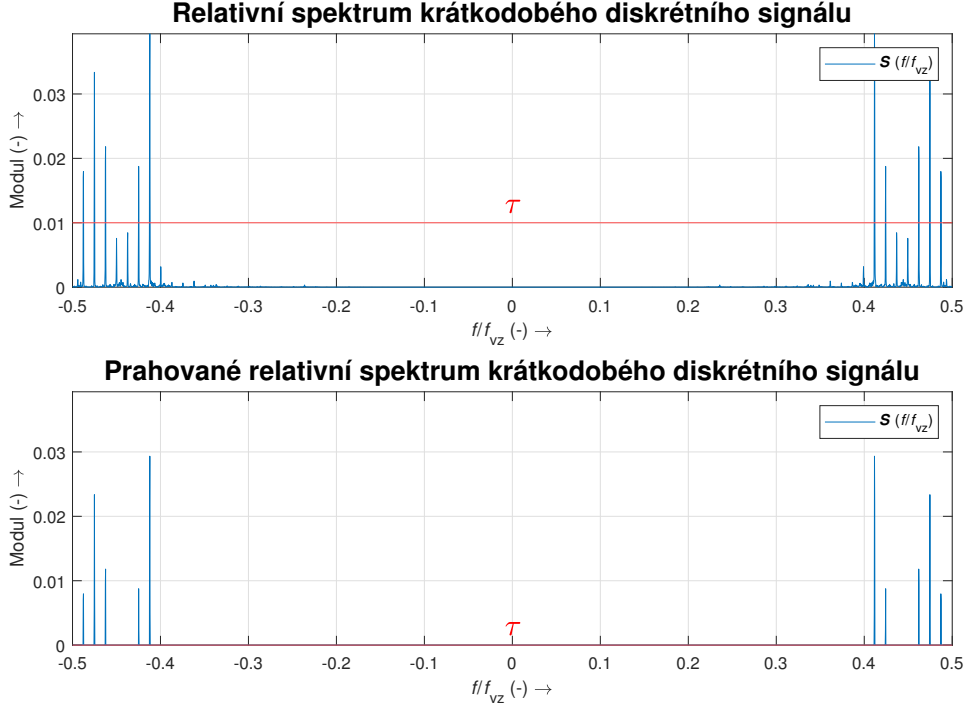
$$f(\mathbf{y}) = \iota_C(\mathbf{y}) = \begin{cases} 0, & \text{pokud } \mathbf{y} \in C, \\ +\infty, & \text{jinak,} \end{cases} \quad (1.10)$$

pak se **proximální operátor** stává **projekcí** na množinu C [17]. Indikátorová funkce $\iota_C(\mathbf{y})$ je matematickým vyjádřením příslušnosti prvku do množiny a nabývá pouze dvou hodnot. Slouží tedy jako penalizace, která hodnotu proximálního operátoru buď nemění nebo zvyšuje, čímž zabraňuje opuštění množiny přípustných řešení. Kromě toho, že proximální operátor minimalizuje funkci $f(\mathbf{x})$, disponuje navíc důležitými vlastnostmi pro iterativní řešení minimalizačních problémů. Proximálním operátorem ℓ_1 normy, který se v inpaintingu využívá, je takzvané měkké prahování (angl. *soft-thresholding*), které pro komplexní koeficienty definujeme jako

$$\text{soft}_f(\mathbf{y}) = \begin{cases} 0, & \text{pokud } |\mathbf{y}| \leq \lambda, \\ e^{j \cdot \arg(\mathbf{z})}(\mathbf{y} + \lambda), & \text{pokud } |\mathbf{y}| < -\lambda, \\ e^{j \cdot \arg(\mathbf{z})}(\mathbf{y} - \lambda), & \text{pokud } |\mathbf{y}| > \lambda, \end{cases} \quad (1.11)$$

kde konstanta λ určuje hodnotu prahu (angl. *threshold*)⁹. Příklad měkkého prahování na vektoru spektrálních koeficientů demonstruje obrázek 1.11.

⁹Pro reálné koeficienty platí $\arg(\mathbf{z}) = 0 \rightarrow e^{j \cdot 0} = 1$ a exponenciální člen tedy z rovnice vypadne.



Obr. 1.11: Demonstrace měkkého prahování vektoru spektrálních koeficientů S .

Syntetizující model

Syntetizující model je jedním z přístupů pro řešení inverzních problémů založených na spektrální řídkosti [16] a pro inpainting byl adaptován jako první [2]. Jeho principem je nalezení malého množství koeficientů (například spektrálních), ze kterých je sestaven výsledný signál. Vychází z rovnice (1.4), místo ℓ_0 používá ℓ_1 normu [18] a indikátorovou funkci ι_Γ :

$$\arg \min_{\mathbf{x}} \|\mathbf{w} \odot \mathbf{x}\|_1 + \iota_\Gamma(S\mathbf{x}), \quad (1.12)$$

kde $\mathbf{x}$ jsou koeficienty signálu, $\mathbf{w}$ je váhovací vektor pro dodatečné přidělení vah jednotlivým koeficientům a ι_Γ omezuje řešení na přípustná, viz rovnice (1.6). Vliv váhovacího vektoru nebyl v této práci z časových důvodů zkoumán a jeho hodnota byla proto nastavena na vektor jedniček¹⁰. Numerické řešení syntetizujícího modelu (1.12) pro inpainting lze najít pomocí Douglasova–Rachfordova algoritmu (dále DR) [17] pro audio inpainting přizpůsobeného v [2].

¹⁰všechny koeficienty tedy mají stejnou váhu

Pro obě funkce z (1.12) $f_1 = \iota_\Gamma(S\mathbf{x})$ a $f_2 = \|\mathbf{w} \odot \mathbf{x}\|_1$ definuje jejich proximální operátory následovně:

$$\text{prox}_{\tau f_1}(\mathbf{x}) = \mathbf{x} - A(M_R \odot S\mathbf{x}) + A(M_R \odot y) \quad (1.13)$$

$$\text{prox}_{\tau f_2}(\mathbf{x}) = \text{soft}_{\tau \mathbf{w}}(\mathbf{x}) \quad (1.14)$$

Řešení algoritmus hledá iterativně pro parametr τ až do splnění ukončovací podmínky:

$$\|\mathbf{x}^{(i)} - \mathbf{x}^{(i-1)}\|_2 < \epsilon \|\mathbf{x}^{(i-1)}\|_2 \quad (1.15)$$

pro $\epsilon > 0$. Výsledkem algoritmu je vektor koeficientů $\mathbf{x}$, který pro získání výstupního signálu musí být na závěr převeden pomocí operátoru syntézy. Pseudokód DR algoritmu je dostupný z [2].

Analyzující model

Analyzující metody jsou starším přístupem pro řešení inverzních problémů [16]. Přesto byly pro inpainting adaptovány později, než syntetizující model [2]. Na rozdíl od syntetizujícího modelu nepracuje přímo se spektrálními koeficienty, ale hledá takovou reprezentaci signálu, která po převedení do frekvenční oblasti vykazuje řidkost spektrálních koeficientů. Ač zní formulace problému velmi podobně jako pro syntetizující model, přístup k řešení se liší. Vychází z rovnice (1.7), místo ℓ_0 používá ℓ_1 normu a indikátorovou funkci ι_Γ :

$$\arg \min_{\mathbf{z}} \|\mathbf{w} \odot A\mathbf{z}\|_1 + \iota_\Gamma(\mathbf{z}), \quad (1.16)$$

kde $\mathbf{z}$ je vstupní signál, $\mathbf{w}$ je váhovací vektor a ι_Γ omezuje řešení na přípustná, viz rovnice (1.6). Numerické řešení analyzujícího modelu pro inpainting (1.16) lze najít pomocí Chambolova–Pockova algoritmu (dále CP) [2]. Pro obě funkce z rovnice (1.16) $f_1(\mathbf{z}) = \iota_\Gamma(\mathbf{z})$ a $f_2(\mathbf{x}) = \|\mathbf{w} \odot \mathbf{x}\|_1$ definuje jejich proximální operátory následovně:

$$\text{prox}_{\sigma f_1}(\mathbf{x}) = (Id - M_R)\mathbf{x} + M_R \mathbf{y} \quad (1.17)$$

$$\text{prox}_{\sigma f_2}(\mathbf{x}) = \mathbf{x} - \text{soft}_{\mathbf{w}}(\mathbf{x}) \quad (1.18)$$

Algoritmus konverguje k řešení pouze pokud pro parametry τ , σ a normu operátoru syntézy S platí podmínka konvergence $\tau\sigma\|S\|^2 \leq 1$. Díky využití těsných framů, které zachovávají ℓ_2 normu při přechodu mezi prostorem (viz rovnice (1.5)) lze ukončovací podmínku definovat jako časový rozdíl aproximací signálu v po sobě jdoucích iteracích:

$$\|\mathbf{y}^{(i)} - \mathbf{y}^{(i-1)}\|_2 < \epsilon \|\mathbf{y}^{(i-1)}\|_2 \quad (1.19)$$

Výsledkem algoritmu je přímo signál v časové oblasti a jeho pseudokód je dostupný z [2].

1.4 Consensus Equilibrium

Všechny dříve zmíněné přístupy k inpaintingu se v mnoha ohledech liší. Klasické metody řídkosti a autoregrese zavádí předpoklady o vstupním signálu, jako je například jeho stacionarita v mezeře. Moderní metody založené na hlubokém učení tento předpoklad rozvolňují a vychází čistě z kontextu kolem mezery. Toto rozvolnění přináší řadu nových možností jako je doplnění signálu s transientní složkou do mezery, čehož klasické metody nebyly schopné. Moderní metody hlubokého učení tuto možnost mají a podle dostupných zdrojů zvládají rekonstruovat i delší mezery až v řádu jednotek sekund, v čemž opět překonávají limity metod klasických [10, 14]. Pokud bude však doplňovaný úsek stacionární, lze předpokládat, že klasické metody mohou poskytnout lepší řešení, neboť jsou na tento případ vymezeny. Metody hlubokého učení mají mnohem širší škálu přípustných řešení a mohou se tedy u stacionárního úseku od toho optimálního vzdálit.

Pro nalezení optimálního řešení by bylo ideální mít možnost různé metody inpaintingu kombinovat, využít jejich silné stránky a získat výsledek, na kterém se všechny shodnou. Právě na tomto přístupu se zakládá iterační koncept *consensus equilibrium*, který kombinuje různé metody inpaintingu tím, že je využívá jako své komponenty (dále agenty). Jedná se o obecný koncept, který nemusí řešit pouze inpainting, ale lze ho adaptovat i na mnoho jiných problematik restaurace poškozených signálů (například declipping nebo denoising). Agentem může být libovolná metoda pro řešení problematiky splňující určité podmínky, které budou rozebrány později (viz sekce 1.4.3). Z dříve zmíněných metod může být agentem například metoda spektrální řídkosti, autoregresní metoda nebo natrénovaný model neuronové sítě. Pro ekvilibrium není podstatná konkrétní funkce agentů, ale to, že se jedná o zobrazení signálu na (alespoň částečně) rekonstruovaný signál. Jednotliví agenti řeší problém v průběhu iterací odděleně a v průběhu iterací je upravován pouze jejich vstup tak, aby směřoval ke společnému řešení.

1.4.1 Consensus optimization

Consensus equilibrium principiálně vychází ze staršího konceptu *consensus optimization*, který je, podobně jako metody řídkosti, definován jako optimalizační úloha [19]. I zde je cílem minimalizovat účelovou funkci (viz (1.8)), která se tentokrát skládá z různých agentů. Ti řeší problém odděleně, ale postupně musí spět ke společné shodě. Jednotlivým agentům jsou přiřazeny váhy, které jim umožňují měnit vliv na výsledné řešení. Formálně lze *consensus optimization* definovat následovně [20]:

$$\underset{\mathbf{x}_i \in \mathbb{R}^N}{\text{minimize}} \quad \sum_{i=1}^N \mu_i f_i(\mathbf{x}_i) \text{ vzhledem k } \mathbf{x}_i = \mathbf{z}, \quad (1.20)$$

kde $\forall i \in \{1, \dots, m\}$ platí, že účelové funkce agentů $f_i(\mathbf{x})$ jsou konvexní funkce a váhy těchto agentů μ_i musí splňovat následující omezení:

$$\mu_i > 0 \quad \wedge \quad \sum_{i=1}^N \mu_i = 1. \quad (1.21)$$

K nalezení řešení *consensus optimization* lze opět využít *proximální dělení* a proximální operátory [17] s tím rozdílem, že agenti mohou začít s vlastní proměnnou $\mathbf{x}_i$ a musí společně směřovat ke stejnému řešení $\mathbf{z}$ (kterým by pro inpainting byl signál doplněný do mezery). K jeho nalezení lze použít například iterační metodu *Alternating Direction Method of Multipliers* (dále ADMM) [19]. Nevýhodou *consensus optimization* je setrvávající omezení agentů $f_i(\mathbf{x})$ pouze na konvexní funkce. To v obecném případě vylučuje použití inference neuronové sítě jako jednoho z agentů, neboť inferenční funkce zpravidla není proximálním operátorem žádné konvexní funkce. Tento limit překonává právě *consensus equilibrium*, které zobecňuje použití *consensus optimization* i na problémy nespádající do kategorie optimalizace.

1.4.2 Consensus equilibrium pro inpainting

Consensus equilibrium bylo poprvé definováno v článku [20]. Autoři v něm popisují jeho vztah s *consensus optimization* a dokazují, že je jeho zobecněním, neboť při použití pouze proximálních operátorů se z ekvilibria stává optimalizační problém. Zároveň odvozují jeho použití pomocí metody *Plug-and-play* i mimo optimalizaci a definují několik algoritmů pro nalezení jeho řešení. Agent nyní nemusí být pouze popsán svou účelovou funkcí, kterou by bylo potřeba minimalizovat (viz rovnice (1.20)). Místo toho je agent samotný je definován jako zobrazení $F_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ jinak nazvané jako neexpanzivní zobrazení. Může jím být cokoli, co zobrazuje vektor hodnot na jiný vektor hodnot (vstupní signál na výstupní), splňuje podmínky pro neexpanzivního zobrazení a řeší daný problém (inpainting), což za určitých podmínek splňují i inferenční funkce neuronových sítí (viz 1.4.3). Požadavkem zůstává shoda na společném řešení $(\mathbf{x}^*, \mathbf{u}^*) \in \mathbb{R}^n \times \mathbb{R}^{nN}$, které lze popsat jako řešení soustavy rovnic [20]:

$$F_i(\mathbf{x}^* + \mathbf{u}_i^*) = \mathbf{x}^* \quad \text{pro } i = 1, \dots, N, \quad (1.22)$$

$$\bar{\mathbf{u}}_\mu^* = \sum_{i=1}^N \mu_i \mathbf{u}_i^* = \mathbf{0}. \quad (1.23)$$

Vektor $\mathbf{u}$ je složen z vektorů u_i vložených pod sebe, které popisují, jak se daný agent odchyluje od celkového řešení. Jedná se hlavně o formální popis odchylky jednotlivých agentů od společného řešení a v praxi jsou vektory u_i součástí jejich vstupu, o který se předchozí řešení v iteracích mění. Tím se agenti v ekvilibriu udržují v hledání společného řešení. Váhy μ_i umožňují nastavit míru vlivu jednotlivých agentů

na celkové řešení. *Consensus equilibrium* narozdíl od *consensus optimization* nekonverguje striktně k jedinému řešení a tyto váhy určují, ke kterému z nich se přiblíží. Jedním z přístupů pro iterativní nalezení řešení ekvilibria je **Mannova iterace** popsaná v [20]. Tento přístup použili i autoři článku [21] pro declipping zvukových signálů pomocí dvou rozdílných agentů – jeden založený na spektrální řídkosti, druhý na hlubokém učení. Článek popisuje algoritmus UPGLADE pro declipping, ze kterého vychází i algoritmus pro inpainting, který je jedním z výstupů této práce (viz algoritmus 1). Navíc se dotýká důležité problematiky neexpanzivních zobrazení, kterou do ekvilibria zanáší agenti využívající hlubokého učení.

1.4.3 Neexpanzivní zobrazení

Aby šly dříve zmíněné metody pro řešení inpaintingu použít v ekvilibriu jako agenti, musí splňovat podmínky neexpanzivních zobrazení. Zobrazení $F_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$ je neexpanzivní, pokud platí [22]:

$$(\forall x \in \mathbb{R}^n)(\forall y \in \mathbb{R}^n) \quad \|F_i(x) - F_i(y)\| \leq \|x - y\| \quad (1.24)$$

Pokud je agent proximálním operátorem, pak je neexpanzivní. Pro složitější metody je posouzení komplikované a nelze u nich v obecném garantovat konvergenci Mannovy iterace. Posouzení konvergence v obecném případě je nad rámec této práce, a proto bude blíže rozebrán pouze případ, kdy sama metoda proximálním operátorem není, ale ani nereflektuje při inpaintingu obsah mezery. Taková metoda nijak nemusí reflektovat vliv změn ve vstupním signálu v iteracích ekvilibria, v případě, že by v něm měla být použita. Například GACELA při inferenci vychází pouze z okolního kontextu mezery a její obsah považuje za ztracený. Při iteračním volání agenta vytvořeného z frameworku GACELA v ekvilibriu by její inferenční funkce generovala stále stejný obsah. S tím si poradili autoři článku [21], kdy agenta nevytvořili přímo z inferenční funkce. Místo toho volají tuto funkci (označenou jako DNN) pouze jednou a nad jejím výsledkem pak vytvořili proximální operátor funkce *squared error*, který se iteračně s definovaným krokem λ přibližuje k řešení inferenční funkce [21]:

$$F_i(\mathbf{x}) = \lambda \mathbf{x} + (1 - \lambda) \mathbf{d}. \quad (1.25)$$

Vektor $\mathbf{d} = \text{DNN}(\mathbf{y})$ je vypočten na začátku před spuštěním ekvilibria a v průběhu iterací se nemění. Krok λ se s každou iterací ekvilibria zvyšuje a jelikož je takto definovaný agent neexpanzivním zobrazením s proximálním operátorem, jednotlivá řešení obou agentů konvergují ke společnému řešení nazývanému též jako pevný bod.

2 Návrh a implementace

Tato kapitola se věnuje návrhu metody inpaintingu založeného na principu *consensus equilibrium* (dále jen CE) a jejích jednotlivých agentů. Nejprve je z metod popsanych v kapitole 1 vybráno několik vhodných kandidátů pro agenty ekvilibria a jsou analyzována praktická omezení vyplývající z jejich implementace. Tito agenti jsou rozděleni do tří kategorií podle schopnosti reagovat na změny v mezeře a je diskutován jejich předpokládaný vliv na výsledné řešení ekvilibria. Následně je popsána architektura programové implementace navrženého algoritmu.

2.1 Výběr metod a návrh agentů

Z **klasických metod** inpaintingu byly vybrány dvě metody založené na řídkosti spektra (analyzující a syntetizující metoda, viz sekce 1.3 a dvě metody využívající autoregrese (*gap-wise Janssen* a extrapolační metoda, viz sekce 1.1.1). Ze zástupců **moderních metod** hlubokého učení byly vybrány neuronové sítě CQTdiff a GACELA, především díky jejich dobré dokumentaci, dostupným modelům a veřejným výsledkům testování, které ukazují slibné schopnosti obou metod.

Agenti byli pro potřeby práce rozděleni na několik typů podle toho, zda jejich výstup závisí na aktuálním obsahu mezery ve vstupním signálu nebo vychází čistě z jeho nepoškozené části. Pro agenty, kteří zachovávají kontext je mezera tím jediným, co se v průběhu iterací ekvilibria mění. V poslední iteraci je navíc provedena projekce výstupních dat, která změny mimo mezeru potlačuje.

Klasické iterační metody využívající autoregrese a spektrální řídkosti vytváří řešení postupně. V každé iteraci se odhadované řešení zdokonaluje, čímž poskytuje možnost přerušit vyhodnocování po menším počtu iterací, vrátit výsledek a navázat na něj znovu později. Taková vlastnost je v ekvilibriu žádoucí, neboť umožňuje vrácené řešení před novým zpracováním upravit. Agent založený na klasické metodě teoreticky ani nemusí začínat s mezerou obsahující pouze nulové vzorky, ale například již existující odhad jiného agenta. Agent disponující těmito vlastnostmi byl pojmenován jako **iterační**.

Pokud naopak metoda poskytuje výsledek nezávisle na obsahu mezery, protože například vychází pouze z jejího kontextu, dá se předpokládat, že v průběhu iterací bude její výstup stále stejný. Takovou vlastnost má například inferenční funkce sítě GACELA, která generuje výstup bez ohledu na aktuální obsah mezery. Jak již bylo popsáno v teoretické části ekvilibria v rovnici (1.25) pro neexpanzivní zobrazení, agenta pro takovou metodu je nutné obalit proximálním operátorem funkce *squared error*, aby byla zaručena konvergence k pevnému bodu. Agent tohoto typu byl pro potřeby metody CE pojmenován jako **neiterativní** a jeho iterační logiku

tak zajišťuje proximální operátor. Je vhodné zmínit, že ekvilibrium sestavené pouze z agentů tohoto typu není optimální, neboť degraduje výsledné řešení na pouhý vážený aritmetický průměr výstupů jednotlivých agentů.

Výhodou neiterativního agenta je možnost předpočítat výsledek metody předem a vytvořit proximální operátor nad výstupní signálem. V případě UPGLADE lze například výsledek $DNN(\mathbf{y})$ v rovnici (1.25) určit před začátkem první iterace ekvilibria. Pro výpočet v ekvilibriu ani není nutné znát vnitřní logiku metody a výsledek například stáhnout z veřejného repozitáře. Proto pouze pro účely testování vznikl referenční agent jako podtyp neiterativního definovaný pouze svým výstupním souborem. Takového agenta lze testovat jen a pouze na signálech, na kterých byl již dříve samostatně otestován a existuje k nim odpovídající výstup. Referenčního agenta proto nelze použít při testování metody CE na obecném vstupním signálu.

2.1.1 Agenti neuronových sítí

Výše zmíněný referenční agent byl vytvořen hlavně kvůli možnosti otestovat síť CQTdiff, která se v praxi ukázala být velmi paměťově náročná. Její inferenční funkce konzumuje v průběhu vyhodnocování přes 16 GB operační paměti RAM, čímž byl překročen přípustný limit zařízení, na kterém byla metoda CE vyvíjena. Důsledkem bylo vypnutí procesu systémem pro vyčerpání operační paměti několik vteřin po spuštění inference. Síť CQTdiff+, trénovaná na kratších mezerách, vykazuje podobné chování. Kvůli tomuto faktu bylo zvoleno alternativní řešení, které alespoň v omezené míře umožnilo otestovat metodu CQTdiff+ na některých vstupech. Autoři sítě publikovali na svých stránkách¹ výsledky testování CQTdiff+ pro mezery v rozmezí 50 až 300 milisekund na nahrávkách vážné hudby z datasetu MusicNet. Součástí každého experimentu provedeného na konkrétním souboru je stejná sada výstupů, jejichž součástí je:

- originální soubor,
- maskovaný soubor (viz obrázek 2.3),
- výsledek metody CQTdiff+.

Na základě těchto dat lze zpětně rekonstruovat masku, kterou autoři použili při daném experimentu. Podobně jako v případě sítě GACELA je pak vytvořen proximální operátor funkce *squared error* (1.25), který se v průběhu iterací přibližuje k výsledku metody CQTdiff+, jejíž výsledek lze načíst ze souboru.

Na základě tohoto předpokladu vznikl referenční agent pro algoritmus CQTdiff+, který vychází z výsledků autorů [3]. Jeho použití v ekvilibriu však celý experiment limituje pouze na stejné vstupy, které použili autoři [3], a není proto bez plně funkční sítě CQTdiff+ možné testovat i mimo tuto množinu signálů.

¹<http://research.spa.aalto.fi/publications/papers/jaes-diffusion-inpainting/>

Sít GACELA též přináší pro použití v ekvilibriu jednu výzvu. Její inferenční funkce sice není tak paměťově náročná jako CQTdiff+, ale v jejím rámci dochází ke korekci hlasitosti pro zvýšení dynamické rezervy. Tím zaručuje, že signál, který bude doplněn do mezery, nepřekročí dynamický rozsah číselné reprezentace a nedojde ke clippingu. Zároveň není snadné tuto normalizaci vypnout. Pro kvalitní rekonstrukci metody CE je nutné, aby agenti výstupní hlasitost signálu neupravovali, neboť postupné zeslabování hlasitosti rekonstruovaného signálu v iteracích by mohlo způsobit zkreslení výsledků. Pokud některý z agentů kvůli svému vnitřnímu preprocesingu snižuje hlasitost na výstupu, je nutné vzít tento fakt v potaz a provést kompenzaci. GACELA implementovaná jako neiterativní agent volá v rámci ekvilibria svou inferenční funkci pouze jednou a v dalších iteracích poskytuje výstup na základě proximálního operátoru. Byl proto zvolen postup, kdy jsou vstupní signál ekvilibria i výstupní signál sítě GACELA normalizovány podle efektivní hodnoty (RMS) spočítané ze vzorků mimo obsah mezery. Tím je zaručena stejná hlasitost agentů před spuštěním iterace.

2.1.2 Agenti spektrální řídkosti

V rámci této práce byly implementovány a otestovány dvě metody inpaintingu založené na spektrální řídkosti, inspirované článkem [2]. Jedná se o analyzující a syntetizující metodu, které byly detailněji rozebrány v sekci 1.3.

Pro dosažení správné rekonstrukce signálu musí konkrétní algoritmy obou metod vzít v potaz některé vlastnosti Gaborovy transformace (dále STFT), zejména použití Parsevalových těsných framů definovaných v rovnici (1.5). První část vymezuje nutnost zachování identity při převodu signálu mezi prostory, tedy musí platit $\mathbf{y} = S(A(\mathbf{y}))$. STFT a její inverzní forma tento vztah splňují při dodržení několika pravidel. Dopředná i inverzní krátkodobá Fourierova transformace musí mít:

1. stejnou délku segmentu,
2. stejnou délku skoku/míru překrytí segmentů,
3. stejný typ okna.

Vhodnou kombinací pro dodržení těchto pravidel je Hannovo okno se 75% překrytím sousedních segmentů. Při nutnosti použít větší překrytí je potřeba kompenzovat nárůst energie signálu po převedení zpět do časové oblasti. Potenciálním problémem může být i s periodizace krajů signálu. V implementaci PyTorch se těmto problémům dá předejít doplněním signálu nulami na začátku a na konci. Druhá část rovnice (1.5) vznáší požadavek na zachování ℓ_2 normy při přechodu mezi prostory. Jinými slovy je vhodné, aby signál v tomto okamžiku neměnil svou energii, což přináší mnoho praktických vlastností. Při výpočtu ℓ_2 normy signálu pak není nutné přecházet mezi prostory a lze ji ekvivalentně počítat jak v časové oblasti, tak z frekvenčních koeficientů.

Toho lze docílit výpočtem normalizačního koeficientu C , kterým je signál normován při přechodu mezi prostory.

$$C = \sqrt{\frac{N}{hop}}, \quad (2.1)$$

kde N je délka jednoho segmentu a hop je délka skoku v počtu vzorků. Tento koeficient má podobný účel jako koeficienty pro energetickou normalizaci FFT, kdy je spektrum děleno odmocninou z délky signálu. Normalizační koeficient je aplikován na každý segment následujícím způsobem:

```
frames = stft(signal)/(C*norm(win));
signal = istft(frames)*(C*norm(win));
```

Člen `norm(win)` představuje ℓ_2 normu použité okénkové funkce. Díky normalizaci je získán Parsevalův frame, pro který jsou oba algoritmy metod řídkosti odvozené. Obě metody jsou iterační a disponují oproti předchozím agentům výhodnou schopností přerušit vyhodnocování, vrátit mezivýsledek a navázat na něj později opětovným voláním. Jak již bylo řečeno dříve, tato vlastnost je v ekvilibriu žádoucí, protože agenti založení na metodě řídkosti spektra mohou reflektovat měnící se obsah v mezeře a na jeho základě korigovat své řešení do dalších iterací. Obě metody disponují několika hyperparametry, jejichž správné nastavení je klíčové pro dobrou rekonstrukci signálu a jejich testováním se zabývá kapitola 3.2.

2.1.3 Autoregresní agenti

Pro testování v ekvilibriu byly vybrány i dvě převzaté autoregresní metody, použité se souhlasem autora Ing. Ondřeje Mokrého, Ph.D. První z nich je upravená verze klasické Janssenovy metody [1], která místo zpracování signálu po pevně daných časových oknech provádí rekonstrukci přímo nad blízkým kontextem mezery [23]. Takto upravená verze dokáže rekonstruovat úseky dlouhé až 80 ms a její implementace, dostupná skrze veřejný repozitář², byla z velké části převzata. Druhou z nich je metoda extrapolace, která byla též implementována v rámci testování pro článek [1]. Tato metoda též vychází z blízkého kontextu mezery a místo jedné sady autoregresních koeficientů odvozuje dva AR modely odděleně pro levý a pravý kontext mezery (viz 1.1.1. Z obou modelů jsou odvozeny výstupní signály, které jsou do mezery doplněny pomocí křížového prolnutí. Obě implementace částečně vycházejí z původních doprovodných materiálů³ k článku [8].

Pro testování v ekvilibriu byla nakonec použita pouze iterační metoda *gap-wise* Janssen neboť autoregresní koeficienty počítá z kontextu i celého obsahu mezery. Díky tomu dokáže její agent disponovat stejnými vstupně-výstupními vlastnostmi

²<https://github.com/ondrejmokry/InpaintingAutoregressive>

³<http://small.inria.fr/keyresults/audio-inpainting/>

jako agent založený na metodě řídkosti. Klíčová je možnost přerušení rekonstrukce v průběhu iterací a následné pozdější navázání po vrácení mezivýsledku. Extrapoláčnická metoda to neumožňuje, neboť pro rekonstrukci nezohledňuje aktuální obsah mezery, ale pouze její okolí. Pokud by měla být v ekvilibriu použita, pak pouze jako neiterativní agent založený na proximálním operátoru (podobně jako GA-CELA).

2.2 Architektura

Následující sekce slouží k návrhu metody inpaintingu CE, jejích komponent, závislostí a některých metod, které byly implementovány v rámci této práce. Jádrem navrhované metody je inspirováno UPGLADE [21], který vychází z Mannovy iterace a popisuje jej pseudokód algoritmu 1.

Algorithm 1 Metoda CE pro inpainting:

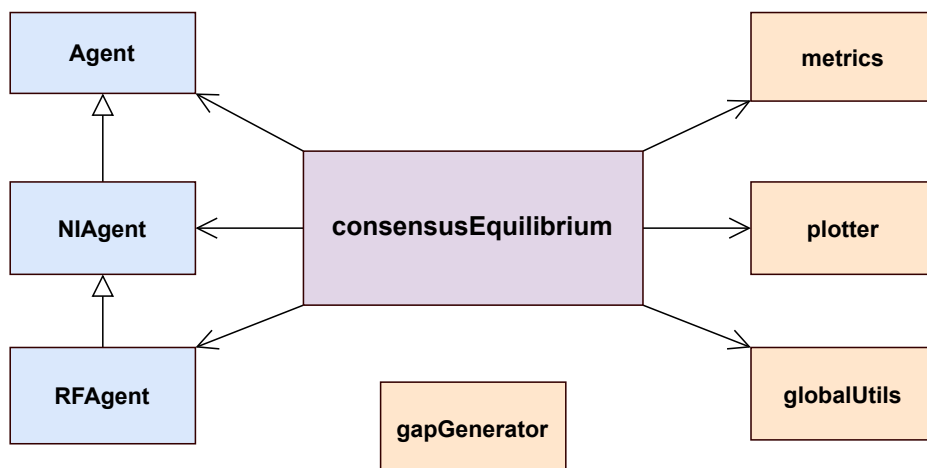
```

1: Input:  $\mathbf{z}_k^{[1]}$   $k = 1, \dots, K$ 
2: Output:  $\text{proj}_\Gamma(\mathbf{z}_{CE})$ 
3: for  $i = 1$  to  $I$  do
4:   for  $k = 1$  to  $K$  do
5:      $\tilde{\mathbf{z}}_k^{[i]} = 2F_k(\mathbf{z}_k^{[i]}) - \mathbf{z}_k^{[i]}$ 
6:    $\tilde{\mathbf{z}}^{[i]} = \sum_{k=1}^K \mu_k \tilde{\mathbf{z}}_k^{[i]}$ 
7:   for  $k = 1$  to  $K$  do
8:      $\mathbf{z}_k^{[i+1]} = (1 - \rho)\mathbf{z}_k^{[i]} + \rho(2\tilde{\mathbf{z}}^{[i]} - \tilde{\mathbf{z}}_k^{[i]})$ 
9:    $\mathbf{z}_{CE} = \sum_{k=1}^K \mu_k \mathbf{z}_k^{[i+1]}$ 

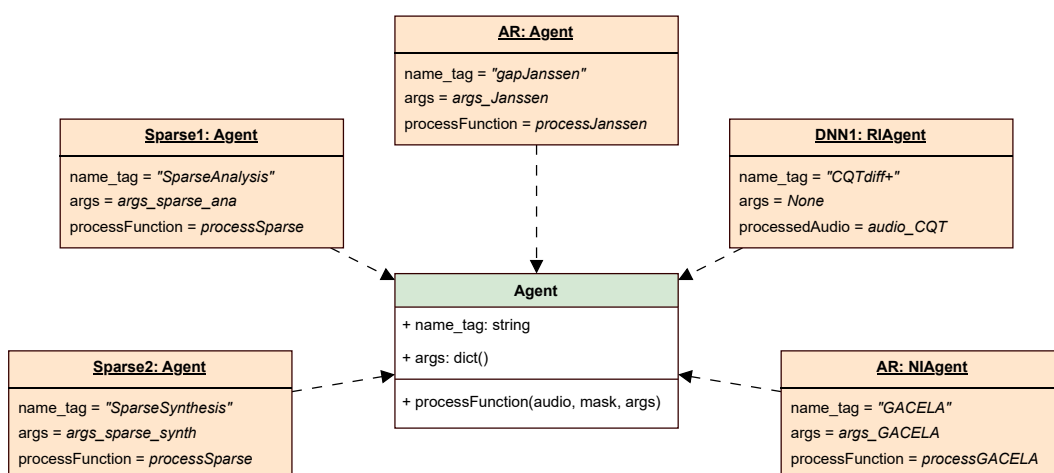
```

Navrhovaná metoda CE obsahuje po vzoru UPGLADE dva nastavitelné hyperparametry, které mají společně s použitými agenty vliv na kvalitu výsledné rekonstrukce. Patří mezi ně počet iterací ekvilibria, který je v algoritmu 1 označen jako I a velikost iteračního kroku ρ . F_k jsou jednotliví agenti a konstanta K určuje jejich počet. Vlivem parametrů na výsledné řešení se zabývá testování v následující kapitole 3.

Diagram na obrázku 2.1 demonstruje základní implementační logiku navrženého algoritmu. Jádrem metody je třída `consensusEquilibrium`, která obsahuje veškerou výpočetní logiku pro inpainting (včetně algoritmu 1), správu agentů a ukládání výsledků do souborů. Využívá i další funkce, které byly odděleny do samostatných tříd. Jedná se hlavně o výpočet metrik v průběhu iterací, vykreslování grafů a další podpůrné třídy. Pro účely testování byl vytvořen i jednoduchý generátor mezer, který dokáže generovat masky s různým počtem různě dlouhých mezer. Roli masky detailněji definuje následující sekce 2.2.1.



Obr. 2.1: Zjednodušený diagram tříd popisující architekturu implementované metody CE

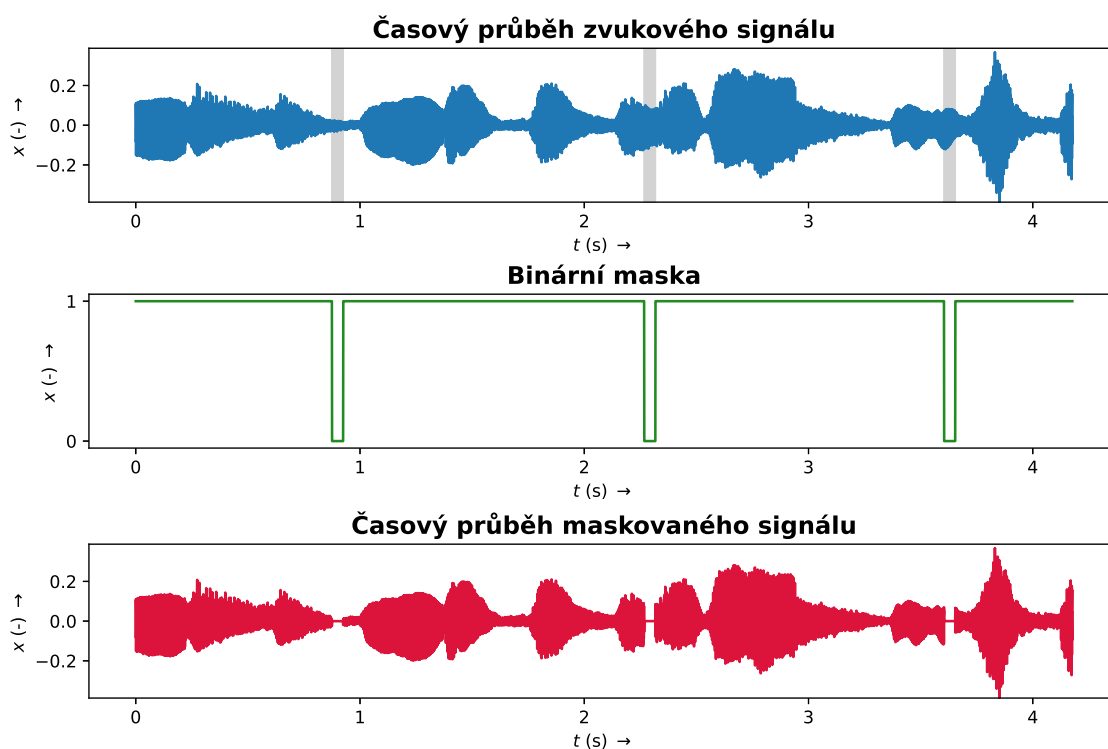


Obr. 2.2: Agent a demonstrace jeho jednotlivých instancí

Architektura algoritmu byla navržena tak, aby byli agenti testovatelní samostatně a zároveň disponovali jednotným rozhraním, které umožní jejich snadné kombinování v ekvilibriu (viz obrázek 2.2). Každý z agentů má uložen identifikátor, slovník obsahující jeho parametry a funkci pro inpainting. Všechny kódy týkající se algoritmu metody CE jsou součástí přílohy B této práce.

2.2.1 Poloha mezery

Inpainting metodou CE vychází stejně jako jeho agenti z předpokladu, že poloha mezery ve vstupním signálu je předem známá. Cílem algoritmu není tuto mezeru hledat, ale pouze do ní na základě znalosti její polohy doplnit vzorky. Pro označení mezer byl použit standardní způsob využívající binární masku, která každému vzorku přiřazuje jeho spolehlivost. Původní nepoškozené vzorky jsou v masce označeny hodnotou 1 (True) a chybějící vzorky v mezeře naopak hodnotou 0 (False). Vynásobením původního signálu s maskou lze získat uměle poškozený signál, u kterého je navíc možné polohu, počet a délku mezer předem určit. Tohoto faktu lze využít při generování různých masek s různým počtem, délkou a polohou mezer. Masky tedy není jen zdrojem poškození, ale i nositelem informace o poloze mezery, a proto je v metodě CE sdílena napříč všemi agenty.



Obr. 2.3: Demonstrace umělého poškození signálu pomocí binární masky.

3 Testování

Tato kapitola se zabývá testováním navržené metody CE a porovnáváním kvality rekonstrukce pro různá nastavení agentů a parametrů rekonstrukce. Cílem je prozkoumat jejich vliv na výsledné řešení, nalézt optimální hodnoty hyperparametrů a porovnat kvalitu rekonstruovaného signálu pomocí objektivních metrik. Ve všech testech byly použity uměle poškozené signály podle předem definované masky (viz sekce 2.2.1), aby bylo možné porovnat kvalitu rekonstrukce jednotlivých algoritmů vůči originálnímu nepoškozenému signálu.

První část testování probíhala na stejném vstupním signálu nahrávky hoboje a fagotu ze souboru `audio/cqt/original/2_1819.wav`. Sledován byl vliv měnící se délky kroku ekvilibria, délky mezery ve vstupním signálu, nastavení vah jednotlivých agentů a počátečních podmínek.

Pro druhou část bylo vybráno několik specifických úloh s cílem otestovat různé vstupní signály, vliv různého počtu mezer a prověřit schopnost ekvilibria rekonstruovat i mezery, které jsou mimo schopnosti některých samostatných agentů v něm.

3.1 Metriky pro testování

Pro měření kvality rekonstrukce bylo vybráno několik objektivních metrik, zaměřených na obecné zvukové signály. Pro účely práce byly nakonec použity tři, které budou blíže popsány v této sekci. Všechny implementace použité v této práci jsou součástí třídy `metrics.py` v kořenové složce projektu, který je součástí přílohy B.

SI-SDR

Hlavní metrikou pro objektivní testování kvality signálu byla zvolena SI-SDR (*scale-invariant Signal-to-Distortion Ratio*). Jedná se o úpravu standardní metriky SDR definované podle [24]:

$$\text{SDR} = 10 \log_{10} \left(\frac{\|\mathbf{y}_{\text{orig}}\|^2}{\|\mathbf{y}_{\text{orig}} - \mathbf{y}_{\text{rec}}\|^2} \right), \quad (3.1)$$

kde $\mathbf{y}_{\text{rec}}$ je rekonstruovaný signál a $\mathbf{y}_{\text{orig}}$ je referenční signál, vůči kterému porovnávání probíhá. Čím je její hodnota vyšší, tím je rekonstruovaný signál podobnější originálu, a proto je žádoucí dosahovat co nejvyšších hodnot. Nevýhodou původní implementace SDR je závislost výsledku na škálování rekonstruovaného signálu, kdy lze pouhým zesílením vstupu dosáhnout lepších výsledků i když obsah signálu zůstal stejný [24]. Tento nedostatek překonává zmíněná varianta SI-SDR, jejíž hodnoty byly měřeny v průběhu iterací ekvilibria i na jeho závěru po dokončení rekonstrukce.

V této práci byla použita implementace převzatá z veřejně dostupné knihovny *auraloss*¹. Pokud bude v práci odkázáno na metriku SDR, je tím míněna její *scale-invariant* verze SI-SDR.

MSL a PEAQ

Multi-resolution spectral loss, zkráceně MSL [26] byla po předchozí konzultaci použita jako náhrada za původně zamýšlenou tradiční metriku PEAQ. Vychází z výpočtu a následného porovnání STFT vstupního signálu spočteného s různými hodnotami rozlišení časového okna a skoku. [26] Čím je hodnota menší, tím je kvalita rekonstrukce vyšší. V metodě CE byla měřena pro výstupní signál i v průběhu iterací.

Visqol

Sofistikovanějším přístupem pro objektivní testování kvality je metrika Visqol, celým názvem *Virtual Speech Quality Objective Listener* [25]. Jeho cílem je simulovat vnímání zvukového signálu člověkem a poskytnout tak alternativu pro subjektivní hodnocení kvality. Pro toto hodnocení počítá odhad hodnoty MOS (*Mean Opinion Score*) na škále od 1 do 5 stejně jako při vyhodnocení poslechového testu, kdy vyšší hodnota odpovídá vyšší kvalitě. Původně byl určen pro řeč, ale od verze 3 nabízí i podporu pro obecné zvukové signály o vzorkovacím kmitočtu $f_{vz} = 48$ kHz a bitové hloubce 16 bitů [27].

V této práci byla použita veřejně dostupná implementace, která nabízí API pro Python². Na testovacím zařízení však jeho použití vykazovalo značnou míru nespolehlivosti, při které vyhodnocování algoritmu bez jakéhokoli chybového hlášení skončilo v průběhu výpočtu metriky. Kromě API je po instalaci knihovny dostupný i spustitelný skript pro bash aplikovatelný přímo na zvukové soubory. Ten byl proto zvolen jako výchozí řešení a volán skrze subprocess a *swap file* v Pythonu. Původním záměrem bylo sledovat vývoj metriky i v průběhu iterací, nicméně kvůli výše zmíněným omezením byla sledována pouze po poslední iteraci u výsledného rekonstruovaného signálu. Další nevýhodou je nutnost převzorkování vstupních dat. Při testování byl ve všech testech použit vzorkovací kmitočet $f_{vz} = 22,05$ kHz, a proto musely být výstupní signály před výpočtem převzorkovány.

¹<https://github.com/csteinmetz1/auraloss>

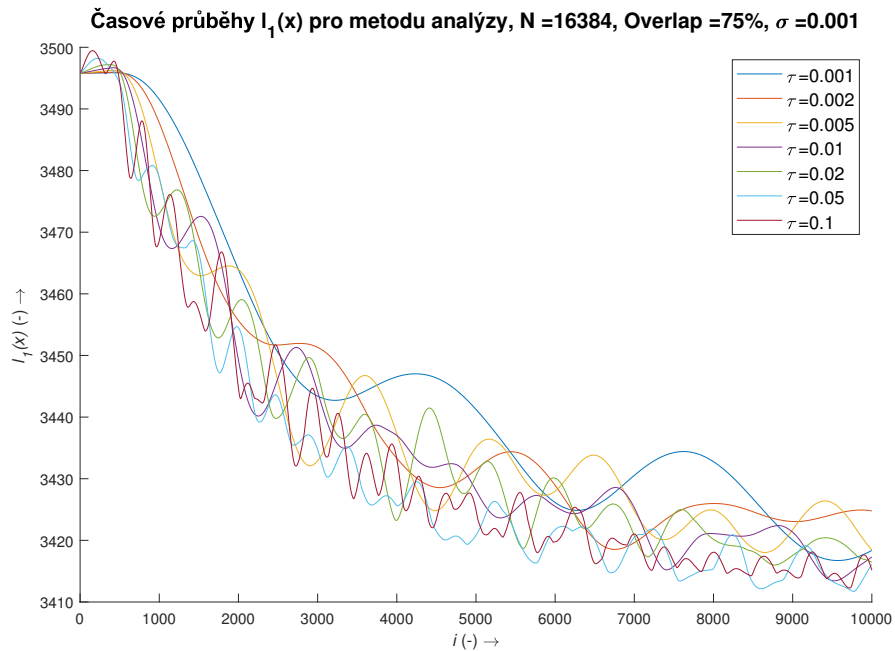
²<https://github.com/google/visqol>

Další metriky

Původně zamýšleným cílem bylo i měření doby rekonstrukce, od kterého bylo nakonec upuštěno. Doba zpracování signálu navrženým algoritmem se v závislosti na počtu iterací, délce vstupního signálu a nastavení agentů pohybuje v rozmezí třiceti minut až jedné a půl hodiny. Za tuto dobu je na běžícím zařízení spuštěna celá řada dalších procesů operačního systému, které by výsledek měření významně zkreslily. Na základě empirického pozorování lze pouze konstatovat, že navrhovaná metoda CE je časově náročná a doba zpracování signálu velmi záleží na nastavení jednotlivých agentů.

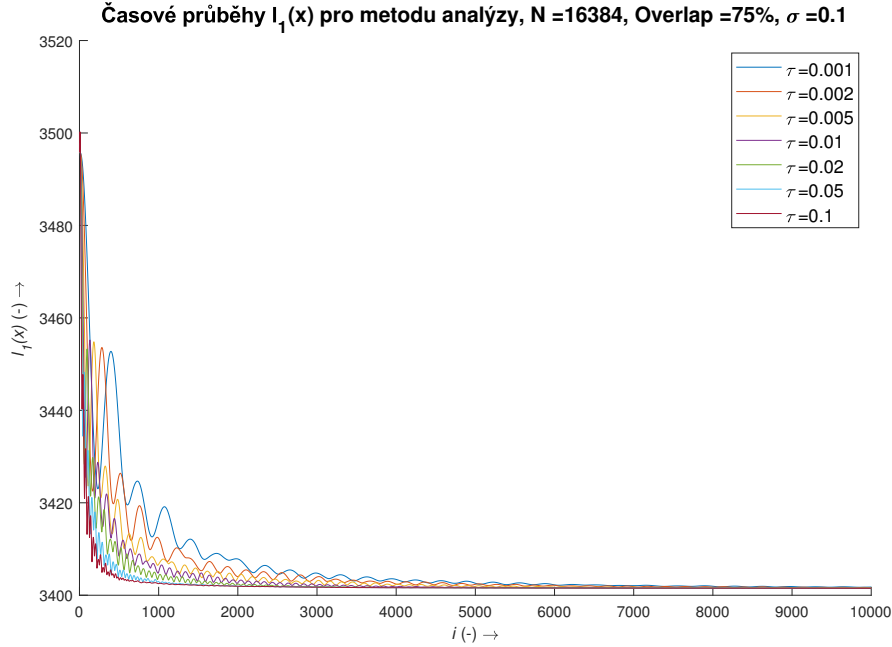
3.2 Testování parametrů řídkostních metod

Protože mají obě metody spektrální řídkosti své vlastní hyperparametry, bylo nejprve provedeno samostatné testování jejich závislosti na výsledné kvalitě rekonstrukce signálu. Ta byla porovnána pomocí objektivní metriky SDR a navíc byl sledován i průběh ℓ_1 normy v iteracích. Testování probíhalo na krátké nahrávce houslí pomocí techniky *hyperparameter sweep*, kdy byl sledovaný parametr postupně inkrementován se současným zafixováním všech ostatních.



Obr. 3.1: Průběh ℓ_1 normy spektrálních koeficientů pro metodu analýzy v iteracích pro mezeru délky 34 ms, fixní hodnotu parametru $\sigma = 0,001$ a jednotlivé hodnoty parametru τ .

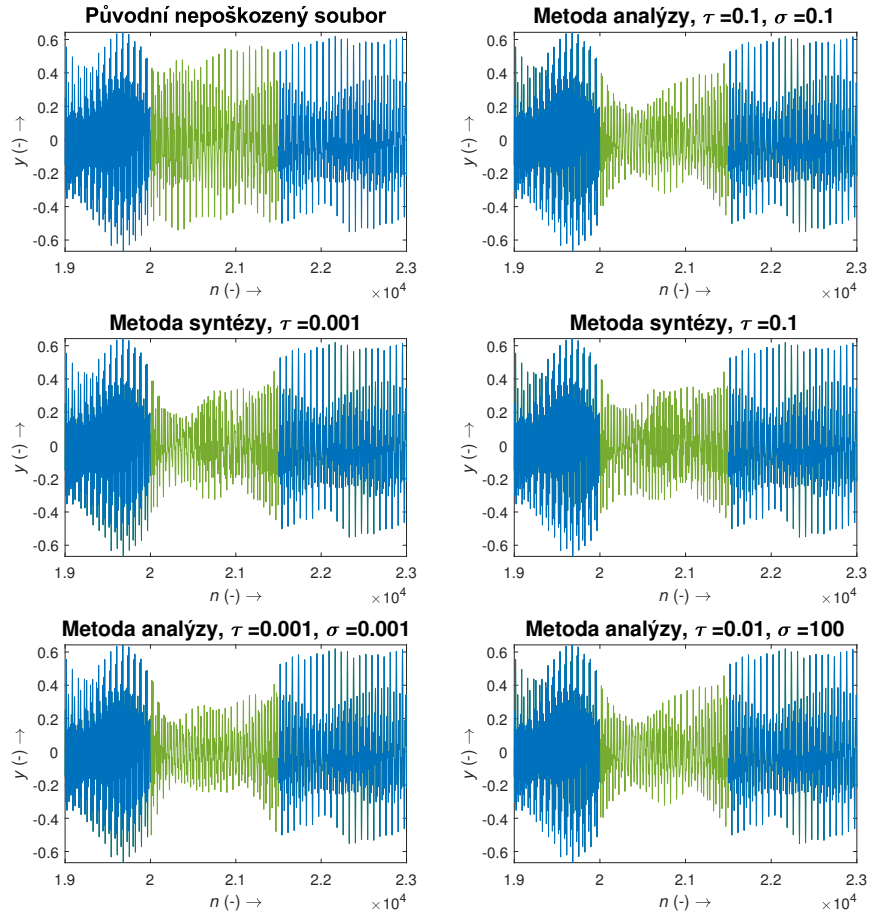
Nejklíčovějšími z nich se ukázaly být konstanty, které definují velikost kroku prahování v rámci iterace. Pro metodu syntézy je to parametr τ a pro metodu analýzy kombinace τ a σ . Tyto konstanty byly popsány v sekci 1.3 teoretické části.



Obr. 3.2: Průběh ℓ_1 normy spektrálních koeficientů v iteracích pro fixní hodnotu $\sigma = 0,001$ a jednotlivé hodnoty parametru τ .

V průběhu $\ell_1(x)$ v iteracích pro metodu analýzy na obrázku 3.1 lze sice pozorovat klesající trend, ale hodnota v čase osciluje a pro zvolený počet iterací nestíhá algoritmus dokonvergovat k hledanému řešení. Z průběhů je patrné, že kvůli příliš nízké hodnotě parametru sigma nestihl algoritmus ani za 10000 iterací dokonvergovat ke stabilnímu řešení pro žádnou z hodnot τ . Průběh na obrázku 3.2 zobrazuje optimálnější volbu kombinací τ a σ . Lze v něm stále pozorovat oscilace, ale pro zvolený počet iterací algoritmus dokonvergoval k hledanému řešení pro většinu hodnot τ . Všechny uvedené kombinace splňují podmínku konvergence (viz 1.3).

Na základě objektivního testování i subjektivního poslechu lze vyvodit, že parametry τ a σ mají významný vliv především na rychlost nalezení řešení. Z průběhů $\ell_1(x)$ v iteracích i časových průběhů signálu na obrázku 3.3 je patrné, že při dostatečném počtu iterací vždy algoritmus konvergoval ke stejnému řešení a volba parametrů σ a τ ovlivnila pouze rychlost konvergence k němu. Kvalita rekonstrukce signálu je postižena pouze v případech, kdy je pro zvolenou kombinaci parametrů stanoven příliš nízký počet iterací a algoritmus za něj nestihne k řešení dokonvergovat.



Obr. 3.3: Vybrané průběhy rekonstrukce pro kombinace krajních hodnot σ a τ . Rozdíly v průběhu časové obálky jsou minimální.

Tab. 3.1: Testování parametru τ v metodě analýzy pro $\sigma = 0,1$.

τ	0,001	0,002	0,005	0,01	0,02	0,05	0,1
SDR (dB)	17,1826	17,1983	17,2038	17,2050	17,2056	17,2059	17,2060

Otestována byla i závislost kvality rekonstrukce na délce okna a skoku. Výsledky ostatních testů jsou součástí příloh této práce. Na základě objektivních metrik i empirického pozorování byly stanoveny optimální hodnoty pro rychlost konvergence při počtu 2000 iterací a vzorkovacím kmitočtu $f_{vz} = 22,05$ kHz. Tyto hodnoty zobrazuje tabulka 3.2.

Tab. 3.2: Optimální nastavení parametrů metody řídkosti. N nastavuje délku časového okna a h délku skoku pro STFT.

Parametr	Analyzující model	Syntetizující model
τ (-)	0,01	0,01
σ (-)	0,1	-
N (-)	16384	16384
N (-)	4096	4096

3.3 Kombinace a nastavení agentů

V první části testování bylo potřeba nastavit referenční kombinaci agentů, která byla sdílena napříč experimenty zmíněnými na začátku kapitoly 3. Zváženy byly silné stránky, ale i omezení jednotlivých agentů, které popisuje dostupná literatura.

Největší limity pro testování nastavují agenti hlubokého učení, neboť dokáží zpracovávat pouze signály charakterově odpovídající těm, na kterých byli trénováni. Jelikož natrénování vlastních modelů by bylo časově náročné a nad rámec této práce, bylo potřeba vyjít z již dostupných natrénovaných sítí. CQTDiff+ nabízí jeden model natrénovaný na datasetu MusicNet³, který obsahuje výhradně nahrávky vážné hudby. V nich se objevují například klavír, housle, viola, klarinet, hoboj a další různé nástroje symfonického orchestru. GACELA má modelů k dispozici několik, trénovaných celkem na 4 různých datasetech (viz sekce 1.2.3). Pokud by měli být oba agenti hlubokého učení testováni společně v ekvilibriu, je potřeba zvolit takovou kombinaci modelů, ve které byli oba trénováni na stejných nebo alespoň podobných datech. Tuto podmínku pro dostupný model CQTDiff+ splňují pouze ty modely sítě GACELA, které byly natrénovány na datasetu Maestro. Jeho součástí jsou klavírní nahrávky, které tvoří i část zmíněného datasetu MusicNet. Většina dostupných modelů obou sítí byla trénována na signálech se vzorkovacím kmitočtem $f_{vz} = 22,05$ kHz, který byl proto zvolen jako výchozí při testování metody CE.

GACELA má veřejně dostupné pouze modely natrénované pro 3 fixní délky mezery – 375, 750 a 1500 ms. Její nevýhodou je již zmíněná neschopnost adaptivity na jinou délku bez nutnosti přetrénovat stávající model. Nejkratší délka mezery, pro kterou je dostupný natrénovaný model, stále značně překračuje hranice rekonstrukčních schopností, které uvádí dostupná literatura pro metody spektrální řídkosti a autoregrese [2, 23]. Pro větší relevanci při porovnání výsledků je vhodné vybírat kratší délky mezer, aby i jednotliví agenti samostatně byli schopni provést rekonstrukci dostatečné kvality. Proto byla pro první část testování vybrána síť CQTDiff+ jako zástupce metod hlubokého učení, neboť její natrénovaný model je více přizpů-

³<https://www.kaggle.com/datasets/imsparsh/musicnet-dataset>

soben omezením klasických metod inpaintingu. Kvůli dříve zmíněným hardwarovým omezením testovacího zařízení (viz sekce 2.1.1) nebylo možné použít její inferenční funkci k vytvoření plnohodnotného agenta. Místo toho vznikl z výsledků oficiálního testování autorů referenční agent (viz obrázek 2.2) pro mezery v rozsahu délek 50 až 300 ms, které lépe pokrývají rozsah schopností klasických metod.

Pro první část testování byla zafixována kombinace 3 metod – **analyzující metoda spektrální řídkosti**, **gap-wise Janssen** a **CQTdiff+**. Síť GACELA byla podrobena několika experimentům ve druhé části testování. Z dostupné teorie lze předpokládat, že klasické metody budou schopné dobře rekonstruovat spíše kratší mezery vzniklé ve stacionárním úseku signálu. Při porušení stacionarity nebo zvýšení délky nad 150 ms může kvalita řešení klasických metod výrazně degradovat. Jedním z cílů testování je zjistit, zda ekvilibrium dokáže dospět k dobrému řešení i v případě, že někteří z jeho agentů sami mezeru rekonstruovat nedokáží.

Globální nastavení agentů pro první sadu testů

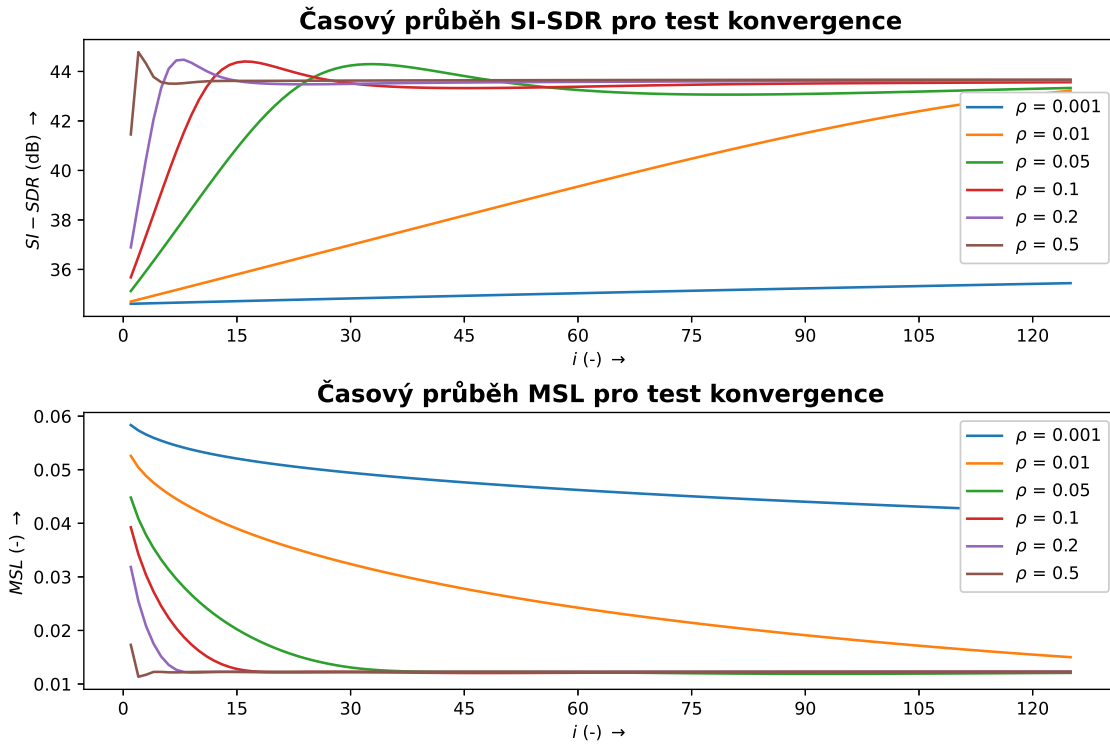
První část testování probíhala na souboru `audio/cqt/original/2_1819.wav`, který je součástí příloh této práce. Pro agenty klasických metod je důležité vzít v úvahu celkový počet iterací, které sami provádí. Agent řídkosti samotný potřebuje zhruba 2000 iterací, aby pro optimálně zvolené parametry τ a σ (viz 3.2) dokonvergoval k řešení. Pokud by byl v ekvilibriu nastaven na 2000 iterací a metoda CE na 20, tak celkový počet iterací provedených tímto agentem v ekvilibriu by činil 40000. Autoři algoritmu UPGLADE [21] toto nastavení použili. Při testování byl řídkostní agent se svým interním počtem iteracím nastaven tak, aby v rámci každé iterace dokonvergoval k řešení. V navrhované metodě CE byl zvolen jiný postup. Agent řídkosti byl počtu iterací přizpůsoben tak, aby celkový počet jeho iterací dohromady s ekvilibiem odpovídal počtu, který by byl nastaven jím samotným pro dokonvergování k řešení. Metoda řídkosti byla v rámci jedné iterace CE nastaven limit na 50 iterací. Počet iterací pro *gap-wise* Janssen byl též snížen na 2 iterace. Při samostatném použití metod řídkosti a autoregrese byl počet jejich iterací zvýšen pro přesnější porovnání výsledků s metodou CE. Počet iterací pro samostatnou metodu řídkosti byl nastaven na 2000 iterací a u autoregrese na 10 iterací.

Tab. 3.3: Globální nastavení parametrů agentů pro první sadu testů.

Analyzující model				GapJanssen		
τ	σ	N	h	délka kontextu	řád	AR model
0,001	0,01	16384	4096	4096	2048	Yule–Walker

3.4 Konvergence

Cílem tohoto experimentu bylo zjistit, jak závisí výsledné řešení na volbě iteračního kroku ρ . I zde byla použita technika *hyperparameter sweep* pro celkem 6 hodnot z rozsahu $\rho \in \langle 0,001; 0,5 \rangle$. Ve vstupním signálu byla vygenerována jedna mezera o délce 50 ms, počet iterací ekvilibria byl nastaven na 125 a váhy všech agentů byly zvoleny rovnocenně, a tedy pro každého agenta platí $\mu_i = \frac{1}{3}$. Parametry agentů odpovídaly hodnotám z tabulky 3.3.

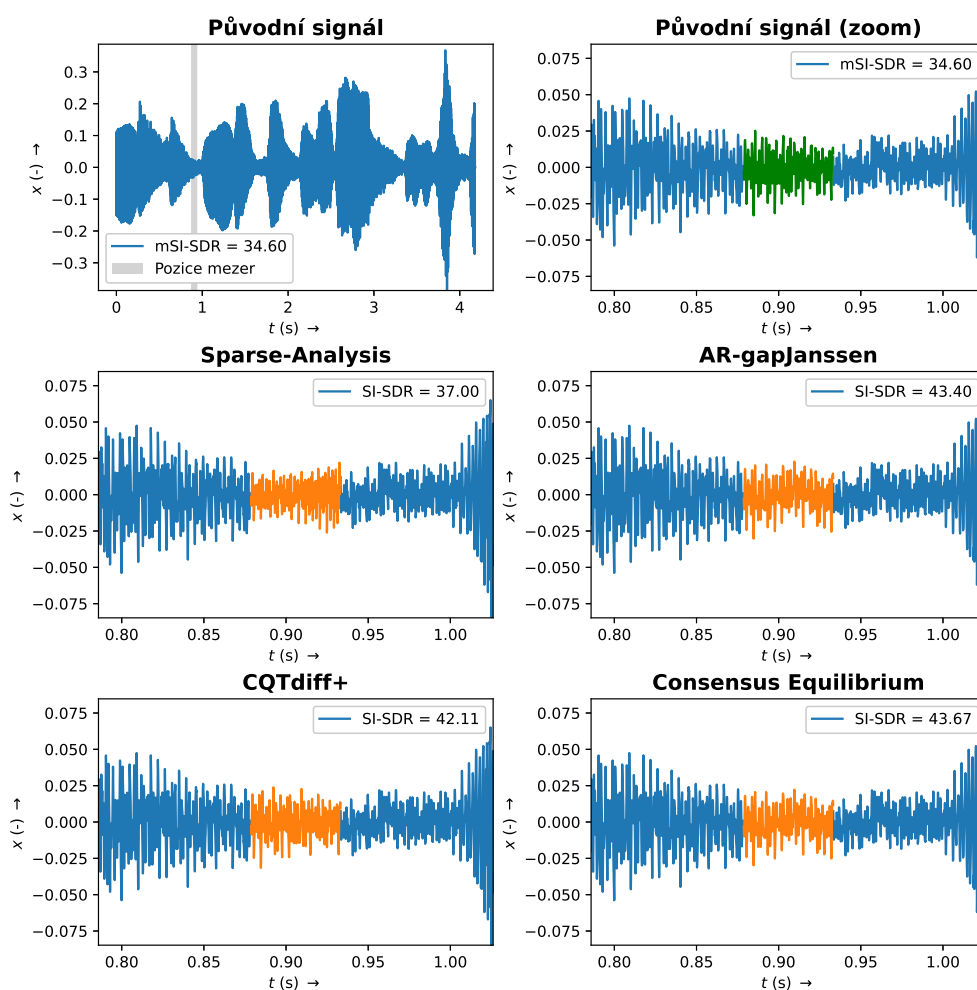


Obr. 3.4: Vývoj SDR výstupního signálu metody CE v průběhu iterací pro test konvergence.

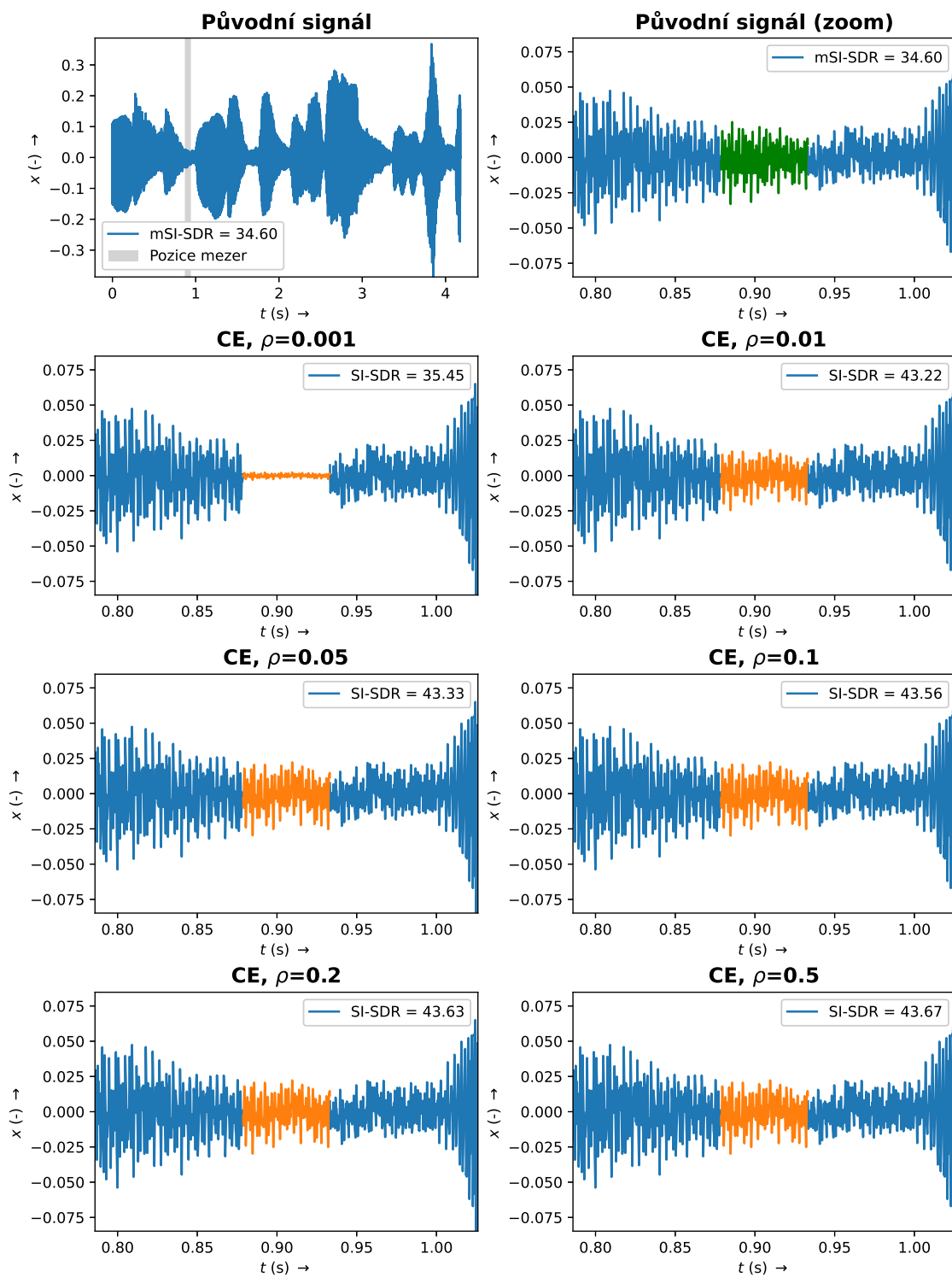
Obrázek 3.4 demonstruje vývoj rekonstruovaného signálu metodou CE v iteracích podle hodnoty SDR. Z něj lze vyvodit, že algoritmus vždy směřuje ke společnému řešení a velikost kroku pouze určuje rychlost konvergence. Tuto hypotézu podporuje i situace pro velikost kroku $\rho = 0,001$, která se pro daný počet iterací ukázala být příliš nízká a algoritmus v tomto případě nestihl dokonvergovat k hledanému řešení. Metoda CE se při počtu 125 iterací ukázala být velmi časově náročnou, a jelikož při tomto počtu a velikosti kroku $\rho = 0,01$ téměř nestihla dokonvergovat k řešení, nelze ani tuto hodnotu považovat za optimální. Rozsah vhodných velikostí kroku pro dostatečně jemný a stabilní průběh rekonstrukce byl na základě testování stanoven na $\rho \in \langle 0,05; 0,2 \rangle$.

Tab. 3.4: Změřené hodnoty objektivních metrik rekonstruovaného signálu pro různé hodnoty ρ . *Mask* označuje vstupní poškozený soubor.

Metrika	Mask	Sparse-Analysis	GapJanssen	CQTdiff+	CE ($\rho = 0,001$)
SDR (dB)	34,602	36,999	43,404	42,108	35,445
MSL	0,060	0,029	0,012	0,009	0,042
Visqol	4,722	4,723	4,724	4,730	4,723
Metrika	CE ($\rho = 0,01$)	CE ($\rho = 0,05$)	CE ($\rho = 0,1$)	CE ($\rho = 0,2$)	CE ($\rho = 0,5$)
SDR (dB)	43,223	43,333	43,565	43,634	43,670
MSL	0,015	0,012	0,012	0,012	0,012
Visqol	4,724	4,724	4,724	4,724	4,724

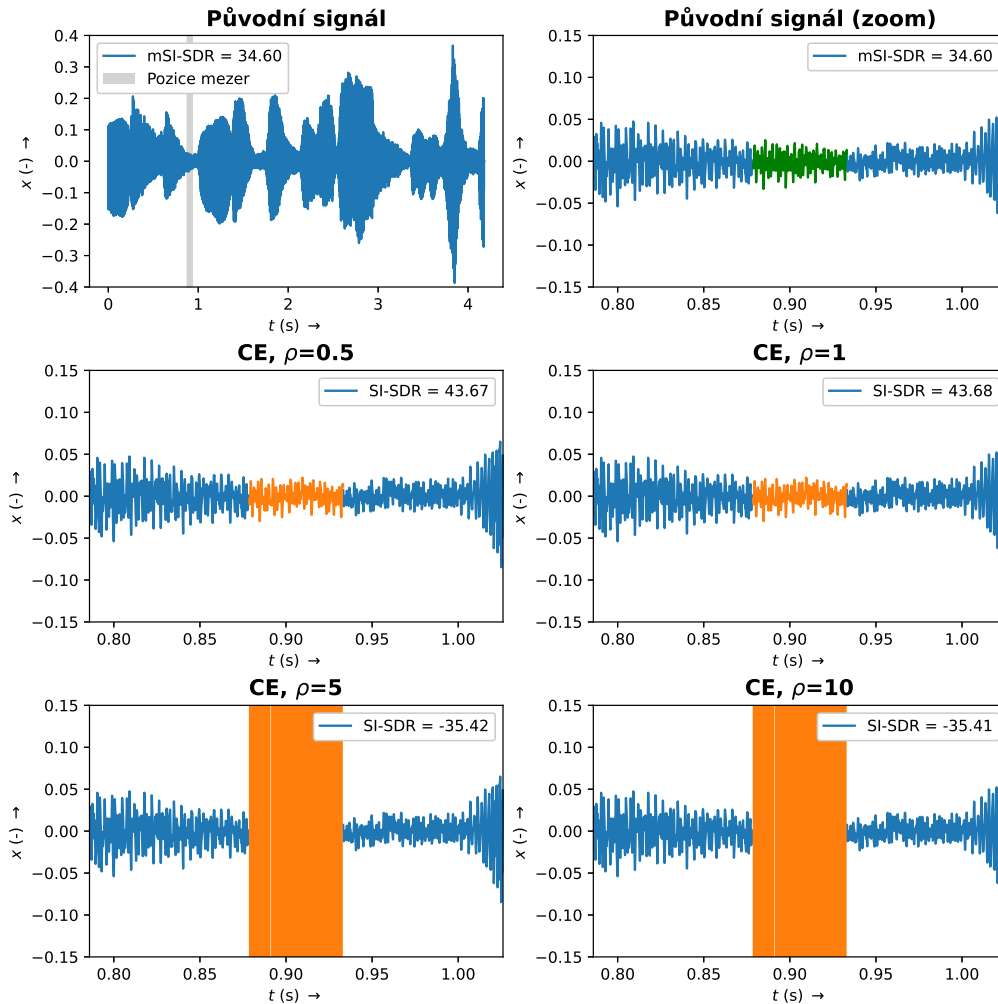


Obr. 3.5: Časové průběhy rekonstruovaného signálu metody CE a jednotlivých agentů pro $\rho = 0,1$ v testu konvergence. Hodnota mSI-SDR od slova „masked“ značí hodnotu SDR pro maskovaný signál.



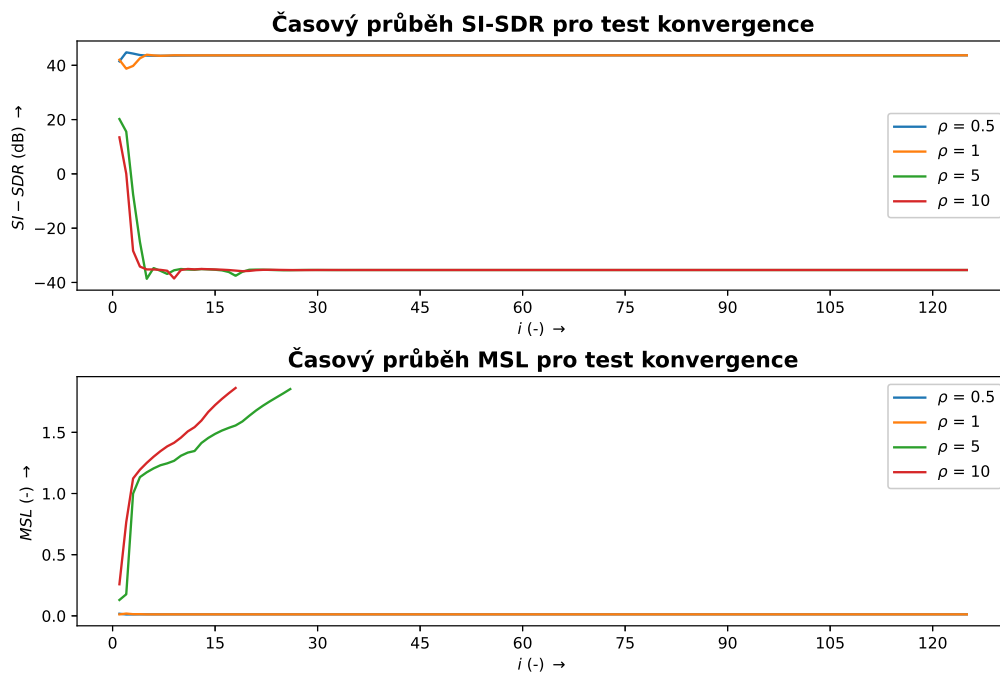
Obr. 3.6: Časové průběhy rekonstruovaného signálu metodou CE pro různé hodnoty ρ v testu konvergence. Pro $\rho = 0,001$ je 125 iterací nedostatečný počet a proto je výsledná energie v mezeře nízká.

Z průběhu 3.4 by se mohlo zdát, že čím vyšší je hodnota kroku, tím menší počet iterací stačí k nalezení řešení. Tato hypotéza je pravdivá jen z části. Větší krok sice urychlí nalezení řešení ekvilibriem, ale příliš vysoká hodnota způsobí nestabilitu algoritmu a nárůst energie v mezeře, jak demonstrují průběh SDR v iteracích (viz obrázek 3.8) i časový průběh rekonstruovaného signálu v mezeře (viz obrázek 3.7).



Obr. 3.7: Časové průběhy rekonstruovaného signálu metodou CE a pro příliš vysoké hodnoty kroku ρ .

V přípěch, kdy metoda CE dokonvergovala k řešení, tedy pro hodnoty kroku v rozmezí $\rho \in \langle 0,05; 0,2 \rangle$, se výsledný rekonstruovaný signál nijak významně neliší ani z hlediska objektivních metrik ani podle subjektivního poslechu. Z výsledků lze proto vyvodit, že parametr kroku ρ má, podobně jako τ a σ u metod řídkosti, vliv pouze na rychlost konvergence k řešení. Pro další testy byla hodnota kroku zafixována na $\rho = 0,1$.

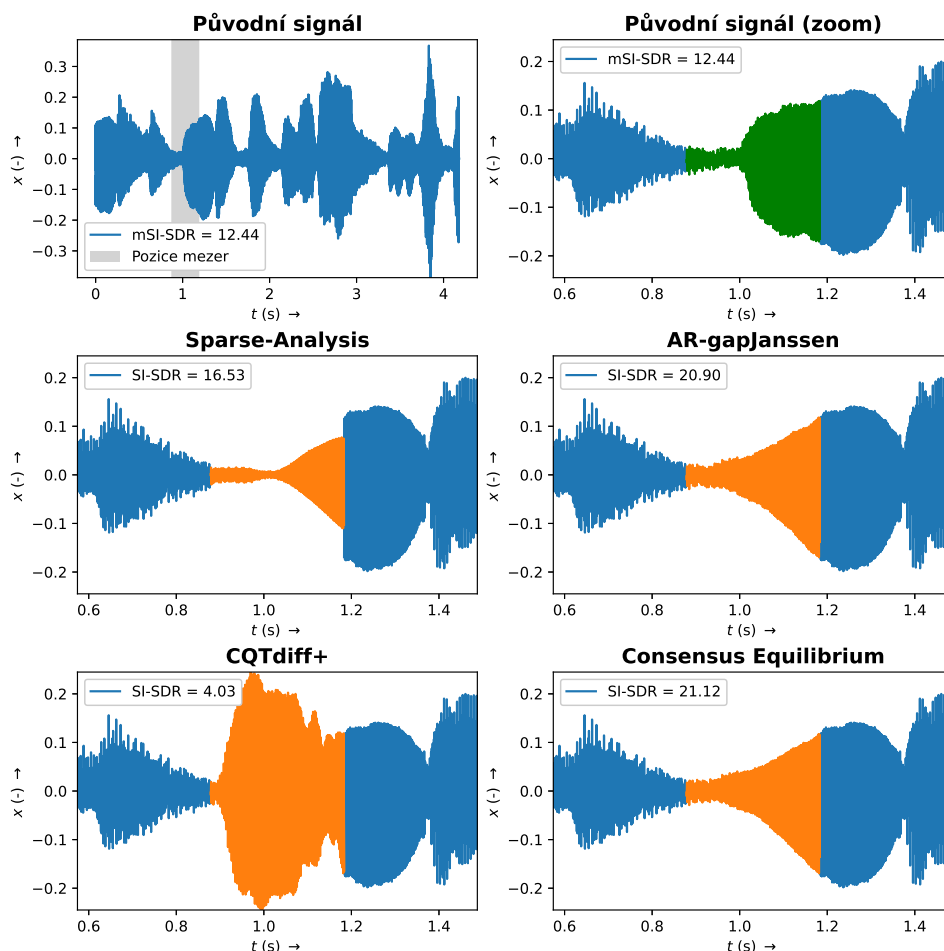


Obr. 3.8: Vývoj SDR výstupního signálu metody CE v průběhu iterací pro příliš velký krok ρ v testu konvergence.

3.5 Délka mezery

Cílem této sady testů bylo zjistit, jaký vliv má délka mezery na schopnosti rekonstrukce ekvilibria. Velikost kroku metody CE byla podle výsledků předchozího testování empiricky stanovena na hodnotu $\rho = 0,1$. Počet iterací ekvilibria byl snižen na hodnotu 75, testovací signál i nastavení agentů zůstaly stejné jako v předchozím testu s jednou mezerou a váhy všech agentů byly nastaveny rovnocenně. Délka mezery byla proměnná a testování probíhalo pro celkem 5 hodnot z rozsahu 50–300 ms. Jednotliví agenti mají svá omezení. Například účinnost klasických metod při inpa-intingu klesá s délkou mezery a nedodržením stacionarity signálu v ní. Předmětem tohoto testování bylo i určit, jak moc své limity propagují do ekvilibria a jestli ekvilibrium dokáže dospět k dobrému výsledku i v případě, že někteří z jeho agentů sami dostatečně kvalitní řešení nalézt nedokázali.

Tabulka 3.5 obsahuje výsledky objektivních metrik pro výsledný rekonstruovaný signál metody CE v porovnání s jednotlivými agenty. Metoda CE dosáhla ve většině případů nejlepších výsledků z hlediska metriky SDR. Výsledek objektivní metriky Visqol však s SDR nekoreluje. Ač jsou rozdíly malé, svědčí v jednotlivých případech ve prospěch samostatných metod *gap-wise* Janssen a CQTdiff+. Překvapivý je především výsledek metody *gap-wise* Janssen, která podle literatury [23] selhává



Obr. 3.9: Časové průběhy rekonstruovaného signálu metody CE a jednotlivými agenty pro mezeru délky 300 ms.

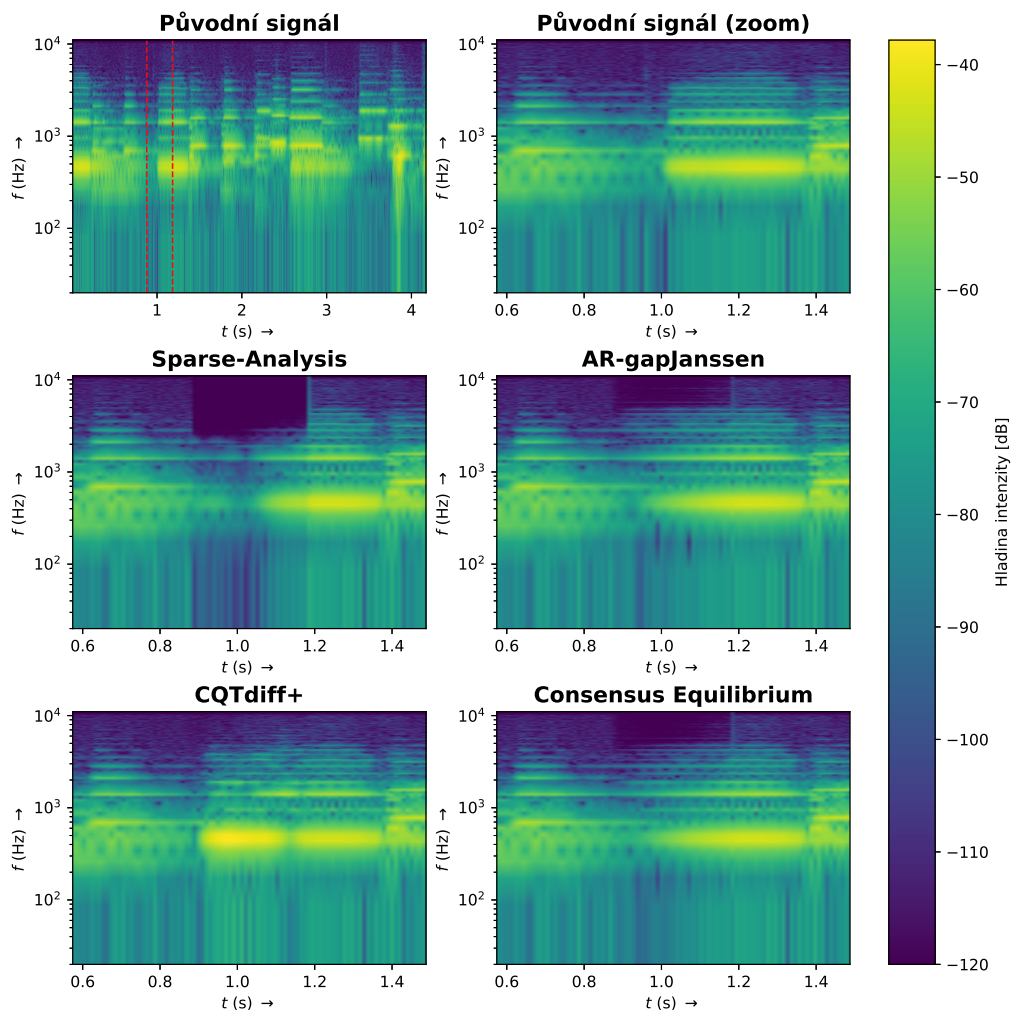
pro mezery delší než 80 ms, zatímco v tomto případě je patrné, že dobře rekonstruuje i mezeru o délce 300 ms. Tato skutečnost je patrná i při subjektivním poslechu. Kvalita rekonstrukce metody řídkosti podle očekávání klesala s rostoucí mezerou s očekávaným poklesem energie uprostřed mezery. Výsledky metriky SDR pro CQT+ vycházely v mnoha případech hůř, především pro delší mezeru. Podle subjektivního poslechu i výsledků metriky Visqol však lze usoudit, že kvalita rekonstruovaného signálu je uspokojivá. Příčinou tohoto rozdílu bylo doplnění nástupu tónu s výrazným dynamickým skokem do mezery. V originálním souboru se sice v oblasti mezery nástup tónu vyskytuje, avšak s menší energií a pozdějším nástupem. Nástup tónu, který CQTdiff+ doplnil do mezery o délce 300 ms je předčasný a z hlediska kontextu hlasitý, což je patrné i při subjektivním poslechu. Metoda CE však přes rovnocenné nastavení agentů výskyt transientního tónu potlačila a výsledný signál je z hlediska tvaru časové obálky velmi podobný výsledku metody *gap-wise* Janssen.

Tab. 3.5: Změřené hodnoty objektivních metrik rekonstruovaného signálu pro různé délky mezery ve vstupním signálu. *Mask* označuje vstupní poškozený soubor.

50 ms	Mask	Sparse-Analysis	GapJanssen	CQTdiff+	CE
SDR (dB)	34,600	37,000	43,404	42,108	43,458
MSL	0,060	0,029	0,012	0,009	0,012
Visqol	4,722	4,723	4,724	4,730	4,724
100 ms	Mask	Sparse-Analysis	GapJanssen	CQTdiff+	CE
SDR (dB)	32,406	33,296	35,521	14,751	33,720
MSL	0,129	0,059	0,036	0,188	0,041
Visqol	4,697	4,695	4,698	4,684	4,698
150 ms	Mask	Sparse-Analysis	GapJanssen	CQTdiff+	CE
SDR (dB)	25,219	22,682	23,098	22,454	23,184
MSL	0,222	0,133	0,098	0,085	0,096
Visqol	4,666	4,682	4,676	4,704	4,676
200 ms	Mask	Sparse-Analysis	GapJanssen	CQTdiff+	CE
SDR (dB)	17,855	21,824	23,208	14,344	23,766
MSL	0,374	0,163	0,107	0,214	0,100
Visqol	4,607	4,652	4,665	4,644	4,663
300 ms	Mask	Sparse-Analysis	GapJanssen	CQTdiff+	CE
SDR (dB)	12,438	16,527	20,904	4,035	21,123
MSL	0,639	0,335	0,137	0,400	0,139
Visqol	4,486	4,536	4,604	4,453	4,597

Ve spektrogramu na obrázku 3.9 lze ve výsledku agenta spektrální řídkosti pozorovat pokles vysokých kmitočtů v mezeře. Ten pravděpodobně vychází z podstaty metody analýzy a jejího proximálního operátoru *clip operator*. Ve vysokých kmitočtech je zde malá koncentrace energie, což potvrzuje i spektrogram původního signálu a právě vysoké kmitočty jsou proto prvním pásmem, které je při postupném aplikování *clip operátoru* potlačeno.

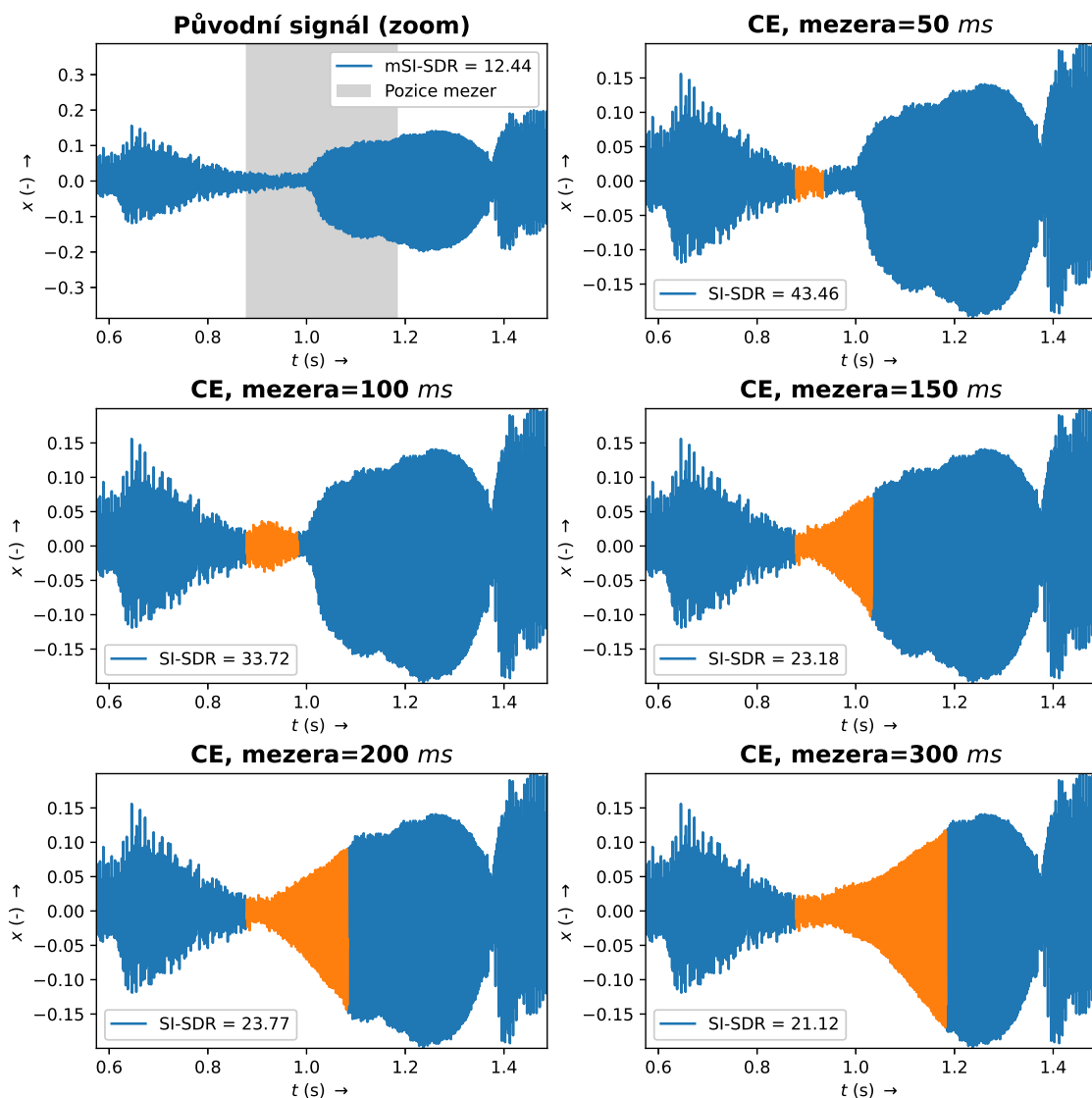
Podobný efekt lze ve spektrogramu 3.9 pozorovat i u agenta autoregresní metody *gap-wise* Janssen. Úbytek energie zde není tak dramatický jako u agenta řídkosti a nastává až v pásmu vyšších středů nad 5 kHz. Tento pokles je pravděpodobně dán nedostatečným řádem autoregresního modelu (viz tabulka 3.3) [23].



Obr. 3.10: Spektrogramy rekonstruovaného úseku délky 300 ms pro jednotlivé agenty. Spektrogramy odpovídají časovým průběhům na obrázku 3.9

Úbytek energie ve vysokých kmitočtech lze pozorovat i ve výsledku metody CE. Pravděpodobně byl do něj zanesen agenty řídkosti a autoregrese. Na výsledku je zajímavé, že přes rovnost agentů dle jejich vah v ekvilibriu neměl agent sítě CQTdiff+ významný dopad na výsledné řešení. To se pro mezeru 300 ms výrazně podobá výsledku autoregresního agenta samotného.

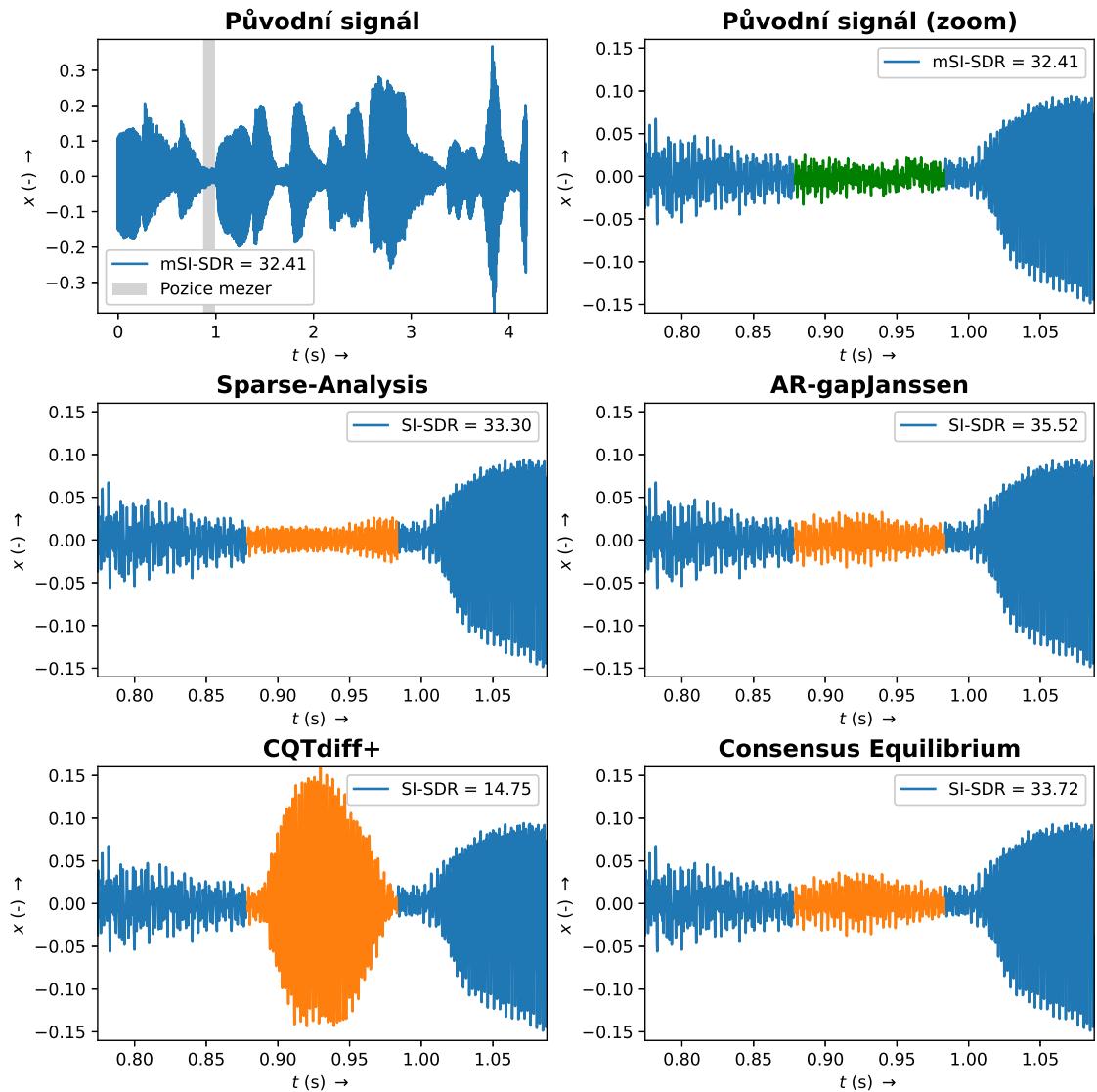
Při subjektivním poslechu je ve výsledku agenta spektrální řídkosti v oblasti na konci mezery slyšet lupnutí, které je pravděpodobně způsobeno hlasitostním skokem, patrným i v časovém průběhu výstupního signálu (3.9). Tento artefakt byl opět zanesen do ekvilibria a i tentokrát v méně výrazné podobě. Způsobovat by jej teoreticky mohla nesprávná práce s daty a indexový posun při aplikaci masky. Pro tento případ to lze vyloučit, neboť ve výsledcích pro mezeru délky 50 ms se tento artefakt nevyskytuje.



Obr. 3.11: Časové průběhy rekonstruovaného signálu metodou CE pro různé délky mezery.

Metoda CE poskytla pro všechny délky mezer velmi dobrý výsledek, jak z hlediska objektivních metrik, tak i subjektivního poslechu. Na obrázku 3.11 je patrné, že dle časového průběhu výsledného signálu měl na výsledek ekvilibria nejzásadnější vliv agent založený na metodě autoregrese. Zajímavá je i situace pro mezeru délky 100 ms, kdy ekvilibrium podle výsledku objektivních metrik neposkytlo nejlepší výsledek a umístilo se až na druhém místě. Velmi pravděpodobnou příčinou této odchylky je agent CQTdiff+. Jeho řešení se pro mezeru délky 100 ms výrazně liší od ostatních agentů, neboť jak demonstruje obrázek 3.12, v ní vygeneroval nový tón, který se v originální zvukové nahrávce nevyskytoval. Při subjektivním poslechu tón působí spíše rušivě, neboť díky své hlasitosti je velmi výrazný a navíc rozbíjí

přirozené tempo skladby. Toto rozhodnutí agenta CQTdiff+ se odrazilo i v jeho horším výsledku metriky Visqol. Metoda CE tento tón potlačila a i přes patrný nárůst energie, který v mezeře způsobil, dospěla k dobrému řešení. Důvodem tohoto úspěchu může být vyšší míra shody agentů autoregrese a spektrální řídkosti na řešení, od kterého se CQTdiff+ liší. Na tomto průběhu je demonstrována první výrazná schopnost metody CE. Nemusí vždy dospět k nejlepšímu řešení, ale dokáže zabránit špatnému řešení. Z porovnání výsledků objektivních metrik z tabulky 3.5 a časových průběhů na obrázku 3.11 je patrné, že kvalita rekonstrukce klesá s délkou mezery.



Obr. 3.12: Časové průběhy rekonstruovaného signálu metody CE a jednotlivých agentů pro mezeru délky 100 ms.

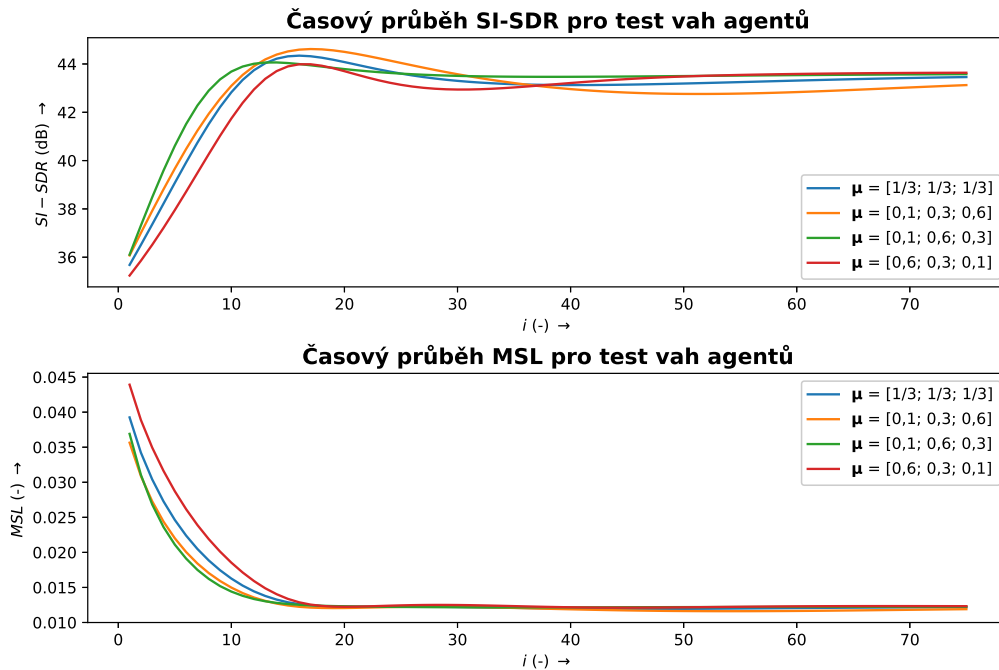
3.6 Váhy agentů

Váhovací koeficienty μ přidělují jednotlivým agentům vliv na výsledné řešení ekvilibria. Cílem této sady testů bylo míru tohoto vlivu prozkoumat. Velikost kroku metody CE byla opět fixována na hodnotu $\rho = 0,1$. Počet iterací ekvilibria byl nastaven na 75, testovací signál i nastavení agentů zůstaly stejné jako v předchozích sadách testů. Testování probíhalo pro mezery různých délek v rozmezí 50 až 300 ms, přičemž byly stanoveny následující kombinace agentů a vah:

Tab. 3.6: Kombinace vah agentů použité při testování.

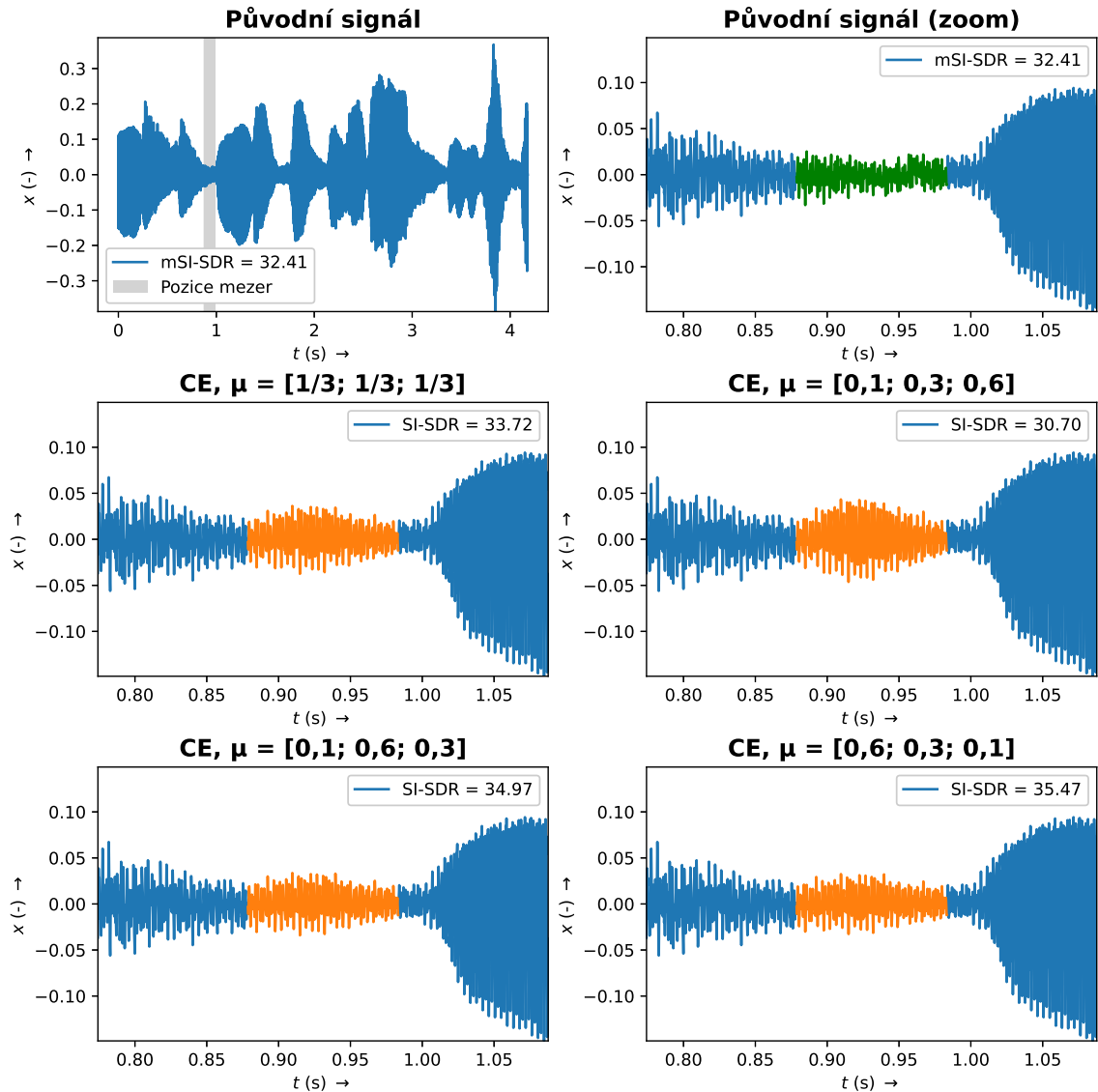
$\mu_1 \rightarrow$ Sparse-Analysis	$\mu_2 \rightarrow$ GapJanssen	$\mu_3 \rightarrow$ CQTdiff+
1/3	1/3	1/3
0,1	0,3	0,6
0,1	0,6	0,3
0,6	0,3	0,1

V této sekci je v grafech zavedena notace, kdy jsou váhy jednotlivých agentů seškupeny do vektoru $\boldsymbol{\mu} = [\mu_1; \mu_2; \mu_3]$. Pořadí jednotlivých hodnot odpovídá pořadí agentů v tabulce 3.6.

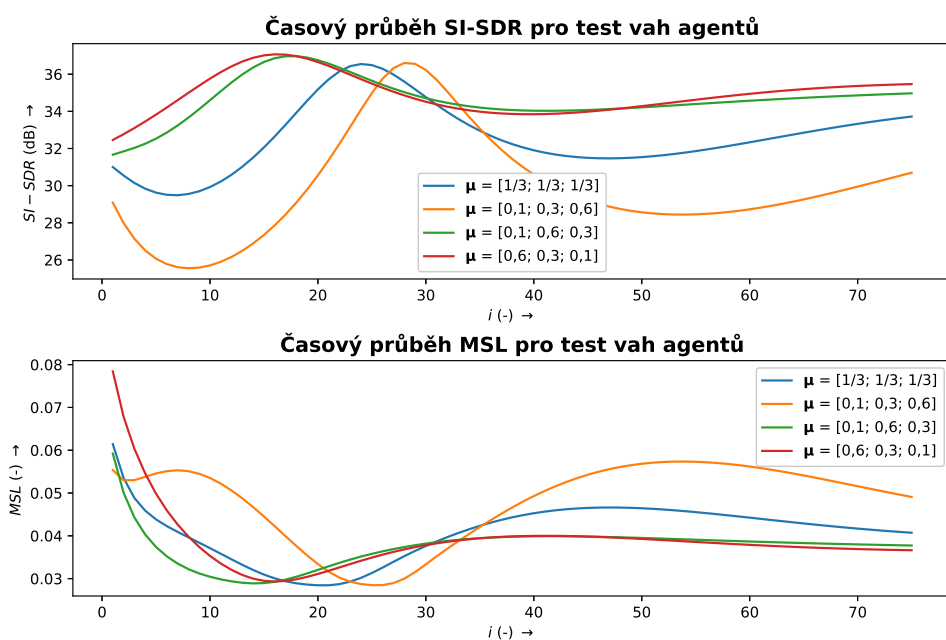


Obr. 3.13: Vývoj SDR výstupního signálu metody CE v průběhu iterací pro mezeru délky 50 ms v testu vlivu vah.

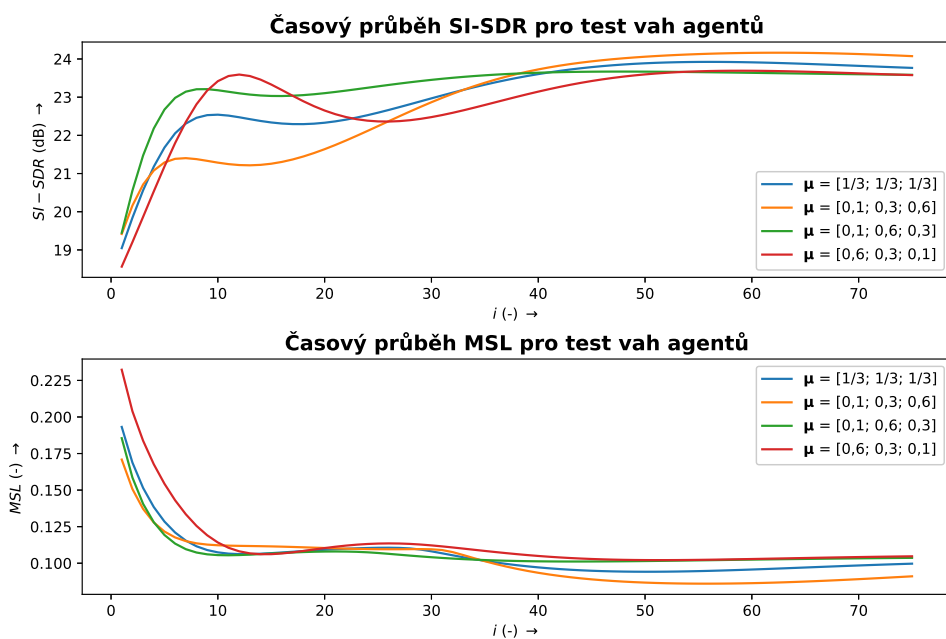
Cílem takto nastavených kombinací bylo učinit jednoho z agentů dominantním a sledovat, jaký dopad bude mít toto nastavení na výsledek metody CE. Z průběhu SDR pro mezeru délky 50 ms v grafu 3.13 vyplývá, že metoda CE dospěla nezávisle na vektoru vah μ k velmi podobnému řešení. Pravděpodobnou příčinou je velká podobnost výsledků jednotlivých agentů, která variabilitu omezuje. Průběh MSL stejně jako v předchozích případech kopíruje trend SDR, a proto mu nebude v tomto testu věnována větší pozornost.



Obr. 3.14: Časové průběhy rekonstruovaného signálu metodou CE pro mezeru délky 100 ms a různé nastavení vah agentů.

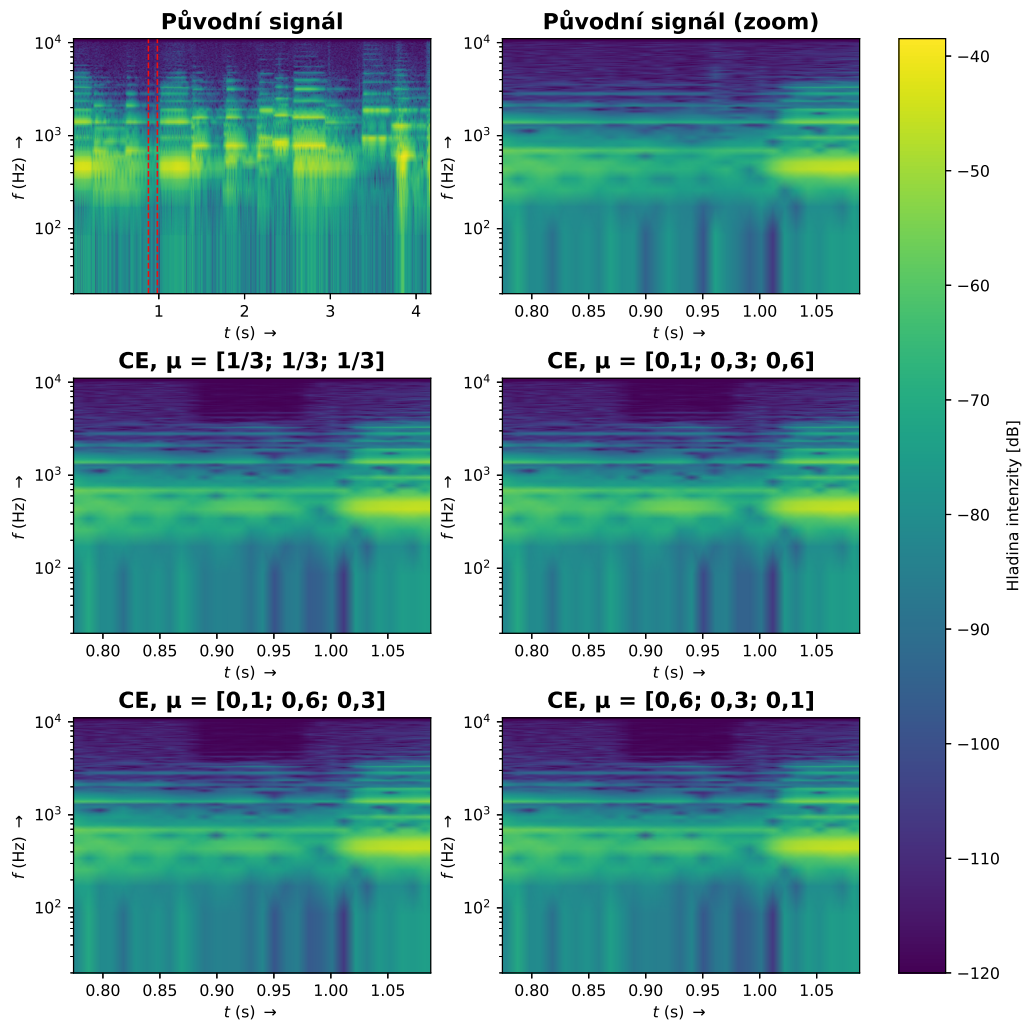


Obr. 3.15: Vývoj SDR výstupního signálu metody CE v průběhu iterací pro mezeru délky 100 ms v testu vlivu vah.



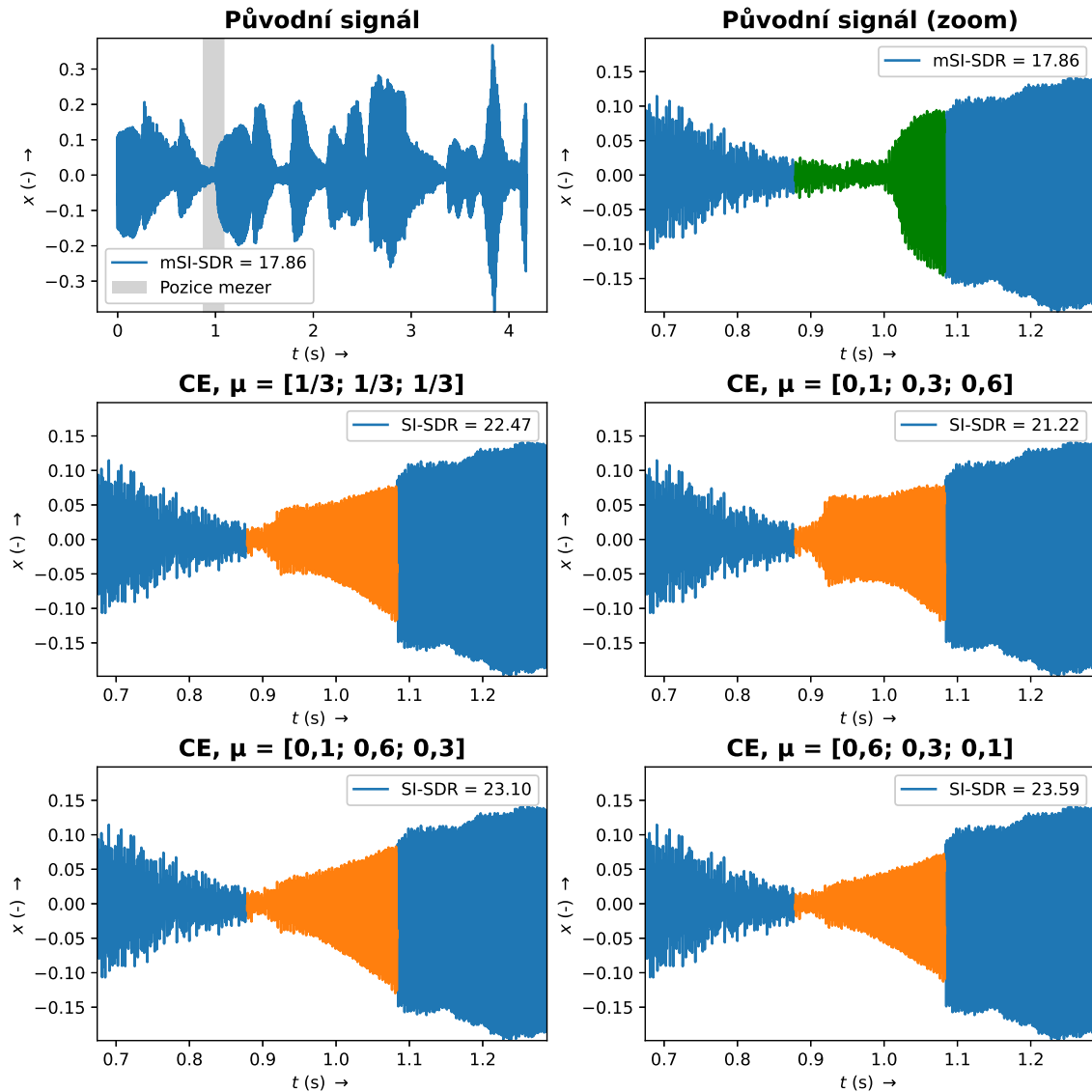
Obr. 3.16: Vývoj SDR výstupního signálu metody CE v průběhu iterací pro mezeru délky 200 ms v testu vlivu vah.

Situace pro mezeru délky 100 ms je odlišná (viz obrázek 3.15). V průběhu hodnot SDR v iteracích lze pro každou kombinaci vektoru μ pozorovat výrazné maximum, vždy v jiné iteraci ekvilibria. Metoda CE ve všech případech dospěla k optimálnímu řešení a po jeho dosažení jej začala degradovat. Z dlouhodobého trendu je patrné, že zlom trval jen chvíli a kolem padesáté iterace došlo ve všech případech průběhu SDR opět k růstu. Na základě toho lze předpokládat, že ekvilibrium při počtu 75 iterací ještě nedospělo k finálnímu řešení. Protože se jedná o stejný testovací soubor se stejnou polohou mezery jako v předchozí sadě testů a liší se pouze nastavením vah agentů, pak jednotlivé výstupy agentů v tomto testu odpovídají časovým průběhům na obrázku 3.12. Oranžový průběh zde pro $\mu = [0,1; 0,3; 0,6]$ odpovídá situaci, kdy největší váha byla nastavena agentu CQTdiff+, jehož výstup se v tomto testu nejvíce lišil od ostatních agentů.



Obr. 3.17: Spektrogramy rekonstruovaného úseku délky 100 ms pro jednotlivá nastavení vah agentů. Spektrogramy odpovídají časovým průběhům na obrázku 3.14

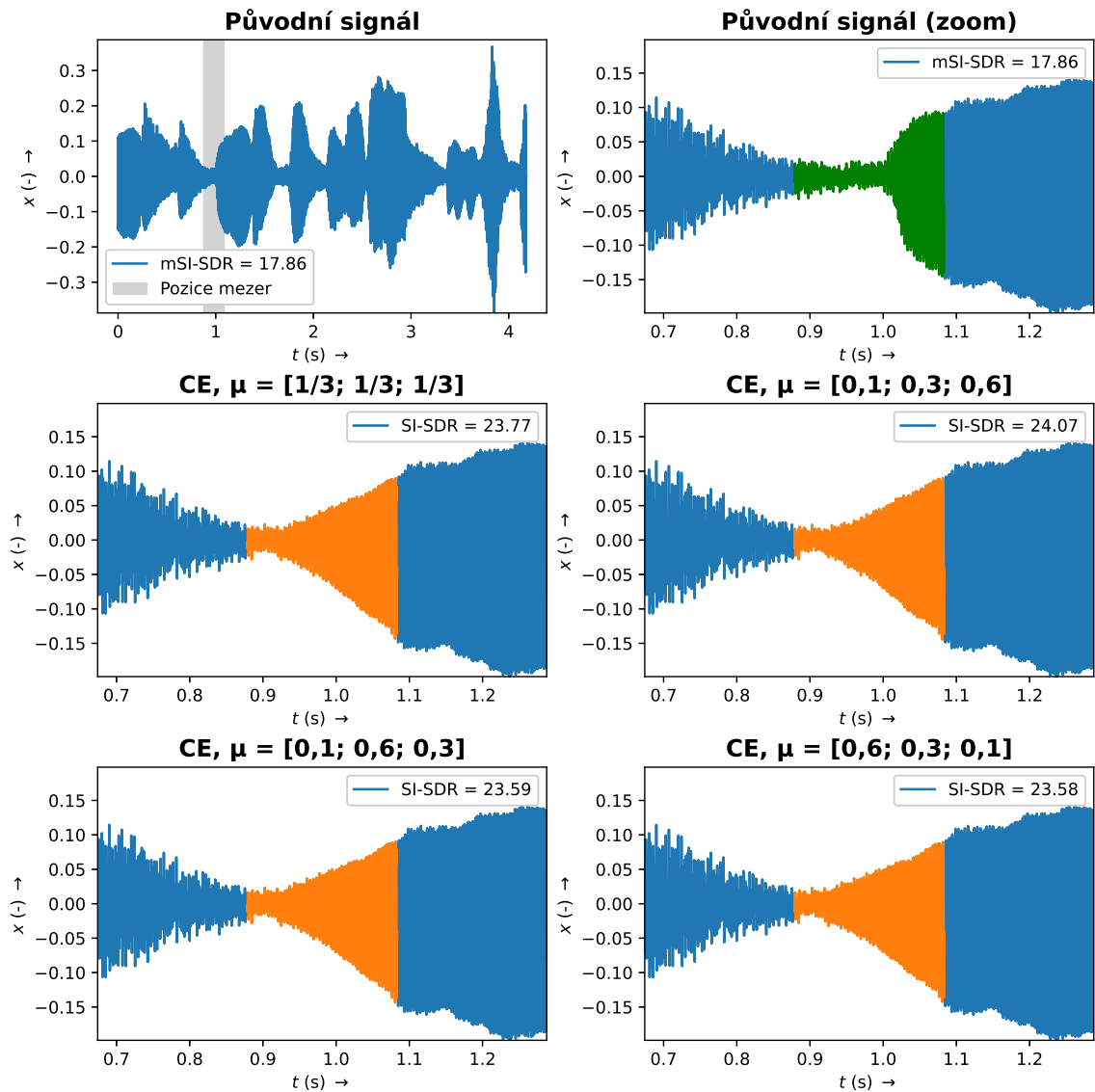
Vzrůstající tendence v poslední iteraci vybízí k hypotéze, zda by se ekvilibrium v dlouhodobém horizontu přiblížilo ke stejnému řešení jako v ostatních případech nastavení vah. Tuto hypotézu podporuje trend pro mezery 200 a 300 ms, neboť přes počáteční výkyvy dospěla metoda CE ve všech případech k velmi podobnému řešení. Průběhy pro 200 a 300 ms jsou velmi podobné, a proto byl přidán pouze graf 3.16 znázorňující SDR v iteracích pro rekonstrukci mezery délky 200 ms.



Obr. 3.18: Časové průběhy rekonstruovaného signálu metodou CE pro mezeru délky 200 ms a různé nastavení vah agentů – 12. iterace

Výsledky této sady testů ukazují, že váhy agentů mají v krátkodobém horizontu vliv na výsledné řešení ekvilibria, nicméně v dlouhodobém horizontu je malý a metoda CE spěla k velmi podobnému řešení nezávisle na vahách agentů (viz obrázek 3.19).

K podobnému řešení směřovalo ekvilibrium i v případech, kdy se jednotliví agenti samostatně ve svých výsledcích velmi lišili. Tento jev je nejvíce patrný u agenta CQTdiff+, jehož výsledek se lišil od klasických metod, ale na výsledek metody CE měl pro všechna nastavení vektoru μ vliv jen v krátkodobém horizontu. Tuto situaci demonstruje obrázek 3.18 znázorňující stav ekvilibria ve 12. iteraci pro mezeru délky 200 ms, kdy dosáhlo SDR pro většinu agentů lokálního maxima. V průbězích obálky signálů jsou vidět změny připomínající novou událost (začátek tónu) generovanou agentem CQTdiff+. Nejvíce se projevila v případě, kdy měl tento agent v ekvilibriu nastavenou nejvyšší váhu.



Obr. 3.19: Časové průběhy rekonstruovaného signálu metodou CE pro mezeru délky 200 ms a různé nastavení vah agentů.

Tab. 3.7: Změřené hodnoty objektivních metrik rekonstruovaného signálu pro mezeru délky 200 ms v testu vah agentů. *Mask* označuje vstupní poškozený soubor.

Počátek	Mask	CQTdiff+	GapJanssen	Sparse-Analysis
SDR (dB)	17,855	14,344	23,208	21,824
MSL	0,374	0,214	0,107	0,163
Visqol	4,607	4,644	4,665	4,652
12. iterace	CE, $\mu = [1/3; 1/3; 1/3]$	CE, $\mu = [0,1; 0,3; 0,6]$	CE, $\mu = [0,1; 0,6; 0,3]$	CE, $\mu = [0,6; 0,3; 0,1]$
SDR (dB)	22,474	21,222	23,101	23,592
MSL	0,106	0,112	0,106	0,108
Visqol	4,653	4,656	4,658	4,647
75. iterace	CE, $\mu = [1/3; 1/3; 1/3]$	CE, $\mu = [0,1; 0,3; 0,6]$	CE, $\mu = [0,1; 0,6; 0,3]$	CE, $\mu = [0,6; 0,3; 0,1]$
SDR (dB)	23,766	24,074	23,585	23,577
MSL	0,100	0,091	0,104	0,105
Visqol	4,663	4,663	4,664	4,663

Tabulka 3.7 porovnává výsledky ekvilibria ve 12. iteraci, místě lokálního maxima SDR (viz obrázek 3.16), s finálními výsledky metody CE z poslední 75. iterace. V místě lokálního maxima dosáhlo nejlepších výsledků podle SDR ekvilibrium s nejvyšší vahou nastavenou u agenta spektrální řídkosti. Tento výsledek je překvapivý, neboť agent sám poskytuje v průběhu všech testů nejméně kvalitní řešení. Podle metrik MSL a Visqol zde nejlepší rekonstrukce dosáhlo ekvilibrium v momentě, kdy byl nejdominantnějším agentem nastaven *gap-wise* Janssen. Výsledky odpovídají subjektivnímu poslechu, ve kterém tyto kombinace též dominovaly.

V 75. iteraci dosáhlo ekvilibrium ve všech případech kombinací vektoru μ lepších výsledků objektivních metrik, než ve 12. iteraci, nejlepších v případě, kdy nejvyšší váhu měl nastaven agent CQTdiff+. Lze tedy usuzovat, že vyšší počet iterací je i přes výkyv v průběhu SDR žádoucí a ekvilibrium v něm dosáhne lepších výsledků. Vliv vah agentů se tím ale snižuje a výsledné řešení metody CE je pro všechny kombinace velmi podobné.

3.7 Počáteční podmínky

Cílem poslední sady první části testování je prozkoumat vliv počátečních podmínek na výslednou rekonstrukci metody CE. Za počáteční podmínky lze označit výchozí hodnoty vektorů $\mathbf{z}_k$ pro $k = 1, \dots, K$ definovaných v algoritmu 1. Tyto hodnoty jsou předány agentům jako vstup v první iteraci ekvilibria. Ve všech dosavadních testech byl, jako společná počáteční podmínka všech agentů, nastaven maskovaný signál, neboli:

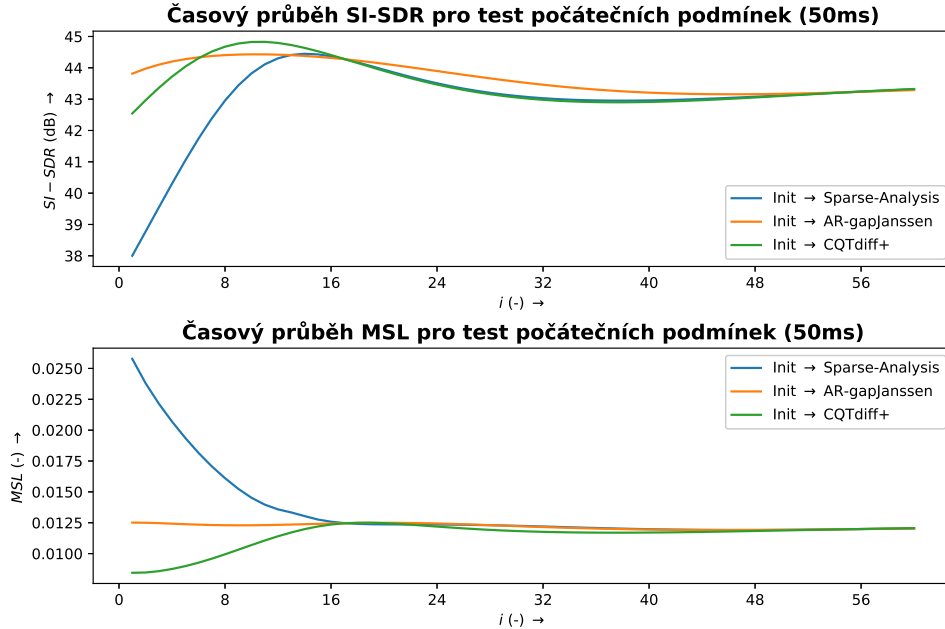
$$\mathbf{z}_1^{[1]} = \dots = \mathbf{z}_K^{[1]} = \mathbf{z}_{\text{in}} \circ \mathbf{m} \quad \text{pro } k = 1, \dots, K, \quad (3.2)$$

kde $\mathbf{z}_{\text{in}}$ je testovaný signál, $\mathbf{m}$ je binární maska definující polohu mezery a $\circ$ je Hadamardův součin vektorů po složkách. Autoři algoritmu UPGLADE vycházejí při svých testech z odhadu agenta neuronové sítě [21]:

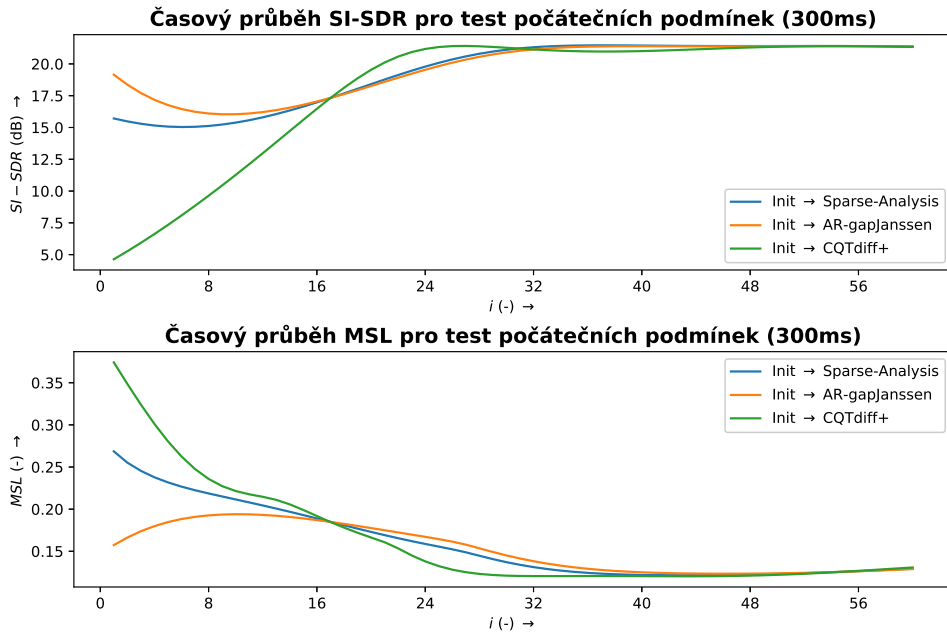
$$\mathbf{z}_1^{[1]} = \mathbf{z}_2^{[1]} = \text{proj}_{\Gamma}(\mathbf{d}) \quad \text{pro } k = 1, \dots, K, \quad (3.3)$$

kde $\mathbf{d} = \text{DNN}(\mathbf{z}_{\text{in}} \circ \mathbf{m})$. Všichni agenti tedy na vstupu v první iteraci ekvilibria neobdrželi maskovaný signál, ale již zpracovaný výstup agenta neuronové sítě. V této práci byly otestovány celkem 3 případy, kdy počátečním odhadem všech agentů metody CE byl vždy výstup jednoho z nich. Protože v iteracích ekvilibria je použito jiné nastavení pro agenty, se kterými by samostatná metoda řídkosti nedokonvergovala k řešení, byl prvotní odhad počátečních podmínek vycházející z agentů řídkosti a autoregrese nastaven stejně jako při odvození porovnání pro samostatné agenty (viz 3.3). Odhad metody řídkosti byl odvozen pro 2000 iterací a odhad metody autoregrese pro 10 iterací.

Velikost kroku metody CE byla opět fixována na hodnotu $\rho = 0,1$. Počet iterací ekvilibria byl nastaven na 75, testovací signál i nastavení agentů zůstaly stejné jako v předchozích sadách testů. Váhy jednotlivých agentů byly nastaveny rovnocenně a testování probíhalo pro mezery délek 50 a 300 ms.



Obr. 3.20: Vývoj SDR výstupního signálu metody CE v průběhu iterací pro mezeru délky 50 ms v testu počátečních podmínek.



Obr. 3.21: Vývoj SDR výstupního signálu metody CE v průběhu iterací pro mezeru délky 300 ms v testu počátečních podmínek.

Tab. 3.8: Změřené hodnoty objektivních metrik rekonstruovaného signálu pro mezeru délky 300 ms ve vstupním signálu. *Mask* označuje vstupní poškozený soubor.

50 ms	Mask	Sparse-Analysis	GapJanssen	CQTdiff+
SDR (dB)	12,438	16,527	20,904	4,035
MSL	0,639	0,335	0,137	0,400
Visqol	4,486	4,536	4,604	4,453
50 ms	CE → Mask	CE → Sparse	CE → AR	CE → CQTdiff+
SDR (dB)	21,123	21,364	21,376	21,334
MSL	0,139	0,129	0,129	0,131
Visqol	4,597	4,600	4,601	4,600

Z vývoje SDR v iteracích pro 50 i 300 ms (viz obrázky 3.20 a 3.21) vyplývá, že nastavení počátečních podmínek má též minimální vliv na výsledné řešení. Dlouhodobý trend SDR v iteracích naznačuje ještě nižší míru vlivu, než při nastavování vah agentů. V šedesáté iteraci je hodnota SDR pro jednotlivé případy téměř totožná.

3.8 Další testy

Druhá část testování zkoumá schopnosti rekonstrukce metody CE na několika dalších modelových příkladech. Cílem bylo otestovat více testovacích signálů, kombinací agentů a poloh mezer.

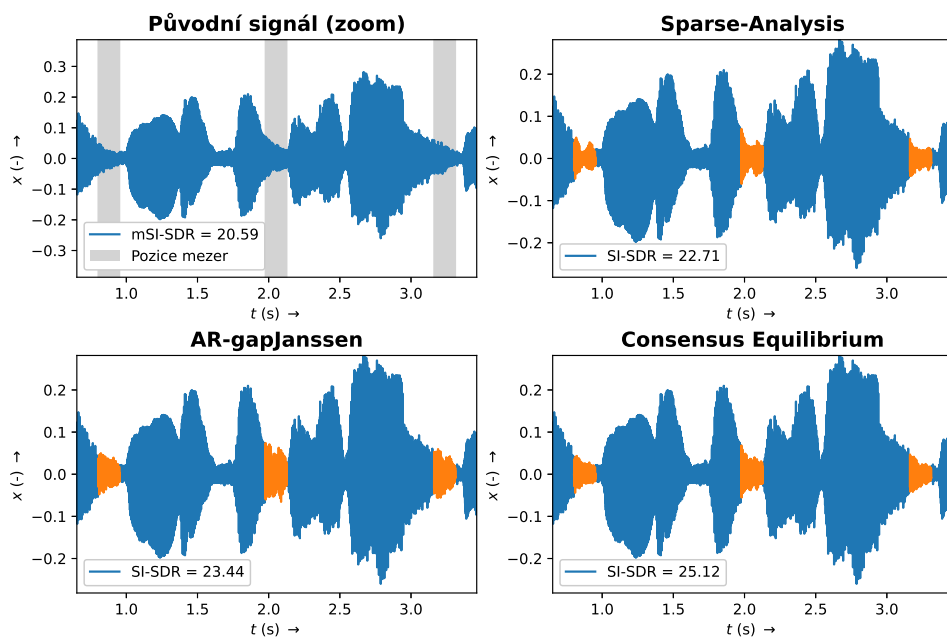
3.8.1 Test řídkosti a autoregrese

V tomto testu byla otestována kombinace pouze klasických metod – agent řídkosti založený na metodě syntézy a agent využívající autoregresi. Oba agenti poskytují podle teorie [2, 23] nejlepší řešení pro krátké mezery do délky 100 ms, i když se v praxi ukázalo, že autoregresní agent je schopný doplnit i delší úseky.

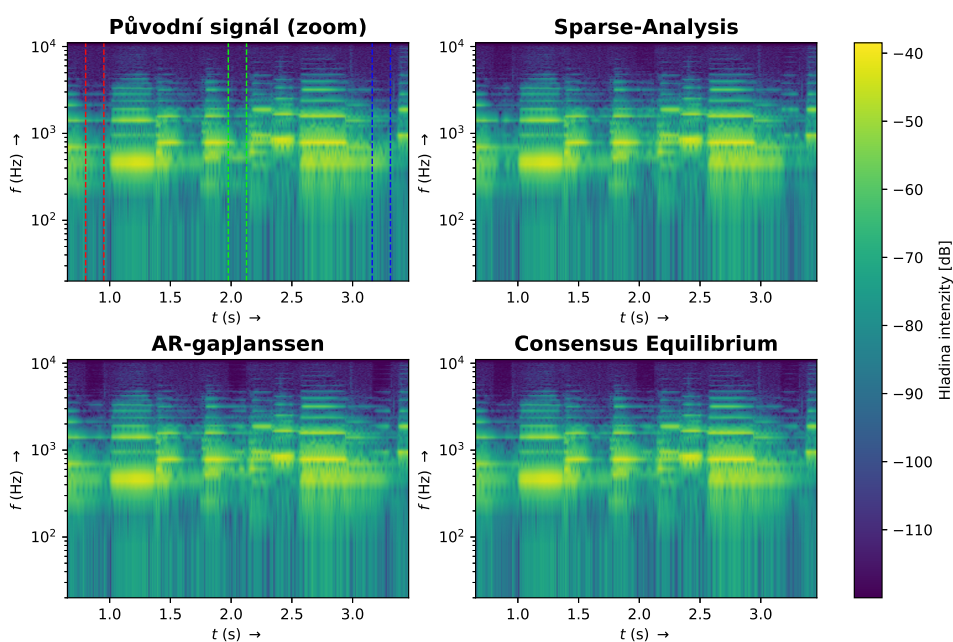
Velikost kroku metody CE byla zvýšena na hodnotu $\rho = 0,2$. Počet iterací ekvilibria byl nastaven na 40 a testovací signál i nastavení agentů zůstaly stejné jako v předchozích sadách testů. Testování probíhalo na stejném vstupním signálu `audio/cqt/original/2_1819.wav` pro mezery délek 50 a 150 ms. Protože obě metody pro krátké úseky generují podobná řešení, bylo mezer v signálu vytvořeno několik pro dosažení větší variability v poškození vstupu. Vzhledem k výsledkům testování počátečních podmínek a vah agentů, byly pro jejich nízký vliv na výsledné řešení nastaveny váhy rovnocenně a jako výchozí vstup ekvilibria byl použit maskovaný signál. Výsledky testování byly zaznamenány do tabulky 3.9. Spektrogram na obrázku 3.23 znázorňuje časově-frekvenční průběhy agentů a metody CE pro mezery délky 150 ms. Oproti předchozím testům zde metoda řídkosti nezpůsobila tak výrazný pokles energie ve vysokých kmitočtech a dokonce v tomto ohledu překonala metodu *gap-wise* Janssen. Tento jev se v malé míře projevil i v ekvilibriu a ve spektrogramu je patrný v oblasti poslední mezery.

Tab. 3.9: Změřené hodnoty objektivních metrik rekonstruovaného signálu pro test autoregrese a řídkosti v ekvilibriu.

50 ms	Mask	Sparse-Analysis	GapJanssen	CE
SDR (dB)	17,902	26,628	29,680	29,919
MSL	0,232	0,056	0,042	0,042
Visqol	4,620	4,686	4,648	4,674
150 ms	Mask	Sparse-Analysis	GapJanssen	CE
SDR (dB)	20,592	22,714	23,437	25,118
MSL	0,608	0,122	0,134	0,118
Visqol	4,552	4,632	4,614	4,622



Obr. 3.22: Časové průběhy rekonstruovaného signálu v testu AR+Sparse pro mezeru délky 150 ms.



Obr. 3.23: Spektrogramy rekonstruovaného signálu v testu AR+Sparse pro mezeru délky 150 ms. Spektrogramy odpovídají časovým průběhům na obrázku 3.22

Tento jev se v malé míře projevil i v ekvilibriu a ve spektrogramu je patrný v oblasti poslední mezery. Nejlepší výsledky v tomto testu poskytla metoda CE, a to jak podle subjektivního poslechu, tak podle výsledků většiny metrik. Pro mezery délky 150 ms je kvalita rekonstrukce dle metriky SDR o 1,5 až 2 dB vyšší než u jednotlivých agentů samostatně. Tento rozdíl, kterého metoda CE dosáhla, je v porovnání se všemi ostatními testy nejvyšší. Výsledky všech metrik spolu ale nekorelují, neboť podle Visqol nejlepší rekonstrukce dosáhla samostatná metoda řídkosti.

Je potřeba zmínit, že ačkoliv byla poloha mezer zvolena náhodně, je pro oba agenty velmi příznivá. Vyskytuje se vždy ve stacionární části signálu v místech, kde dochází k postupnému poklesu amplitudy.

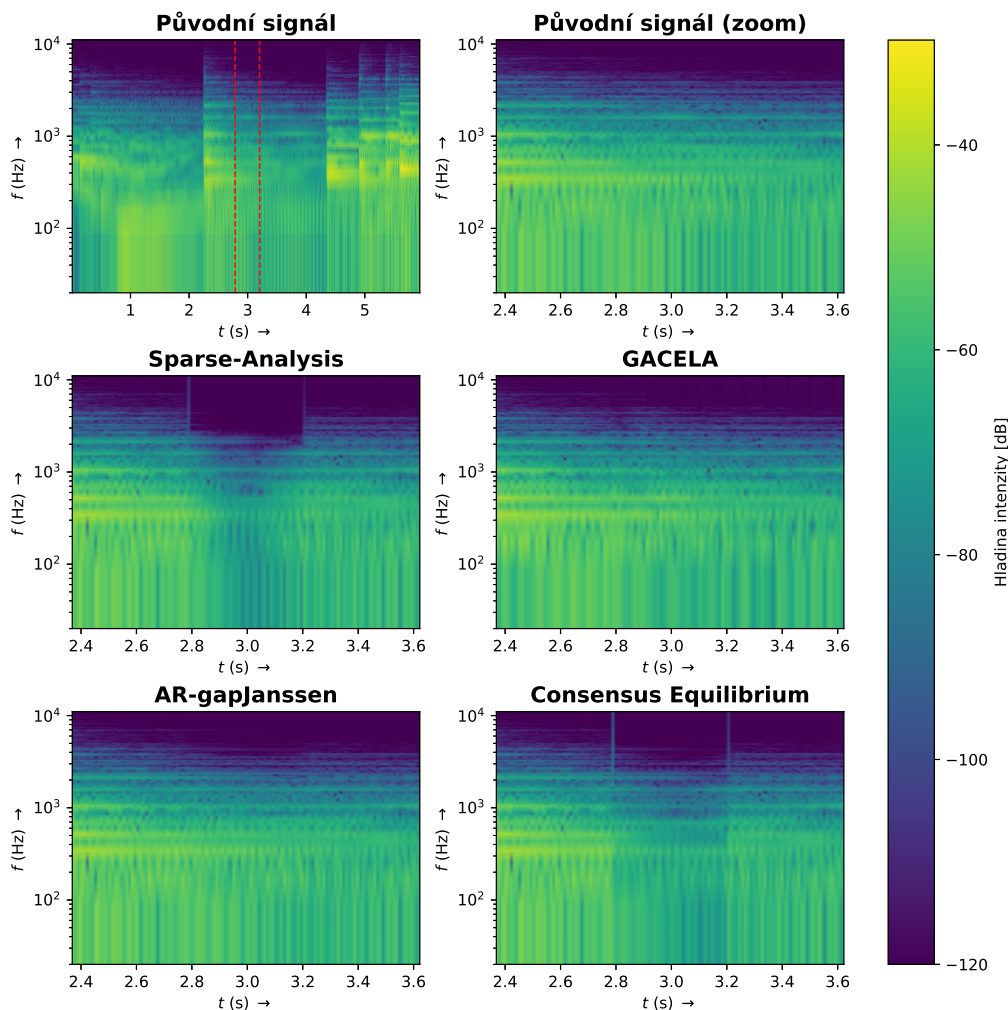
3.8.2 Test sítě GACELA

Tento test metody CE vychází ze stejného nastavení jako testy z první části této práce, pouze s obměnou jednoho z agentů, kdy místo sítě CQTdiff+ byla použita síť GACELA. Poloha mezery byla nyní definována agentem sítě GACELA, která byla trénována na mírně odlišném typu poškození, než všechny ostatní metody použité v této práci. Místo poškození signálu v časové oblasti pomocí binární masky provádí segmentaci a převádí signál do časově-frekvenční oblasti, ve které zahazuje několik po sobě jdoucích časových framů. Pro účely testování byla pro ostatní agenty podle chování sítě GACELA vytvořena ekvivalentní binární maska v časové oblasti, která zahazuje všechny vzorky, se kterými se framy smazané sítě GACELA překrývají. Testování probíhalo na dvou signálech o vzorkovacím kmitočtu $f_{vz} = 22,05$ kHz.

Prvním byl vstupní signál `audio/piano_midi/piano_midi1.wav` pro jednu mezeru délky 371 ms. Pro inferenci byl použit model sítě GACELA označený jako `real_data_240_32_240`, který byl natrénován na nahrávkách klavíru. Původním cílem bylo využít stejný vstupní signál `audio/cqt/original/2_1819.wav` jako v první části testování. Síť GACELA však jeho použití znemožňuje kvůli pevně dané délce mezery a kontextu, který svými požadavky překračuje délku tohoto signálu. Jinými slovy je pro použití v síti GACELA příliš krátký.

Druhým testovacím signálem byla rocková píseň Morning Birds z datasetu FMA⁴, na jehož rockových skladbách byl trénován model označený jako `fma_rock`. Velikost kroku metody CE byla zvýšena na hodnotu $\rho = 0,2$. Počet iterací ekvilibria byl nastaven na 20 a testovací signál i nastavení agentů zůstaly stejné jako v předchozích sadách testů vyjma agenta řídkosti, který stále vycházel z metody analýzy. Agentu řídkosti byly navýšeny hodnoty parametrů τ na hodnotu 0,005 a sigma na hodnotu 0,05. Důvodem pro toto nastavení bylo urychlení testování.

⁴<https://github.com/mdeff/fma>

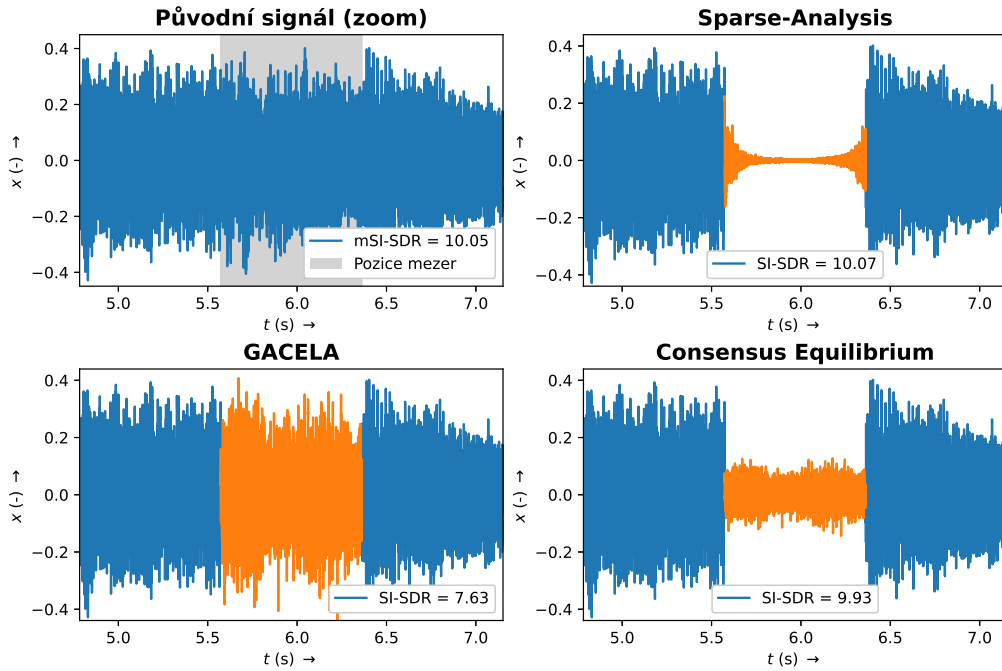


Obr. 3.24: Spektrogramy rekonstruovaného signálu metody CE a jednotlivými agenty v prvním testu sítě GACELA pro nahrávku klavíru.

Použití sítě GACELA v ekvilibriu se ukázalo být velmi problematické kvůli způsobu, jakým manipuluje se vstupním signálem. Při inferenci využívá pouze modulové spektrum signálu a fázi rekonstruuje z výsledného spektrogramu pomocí algoritmu PGHI (viz sekce 1.2.3). Nezachovává však původní kontext signálu mimo mezeru a výsledkem je signál v časové oblasti, který je podle subjektivního poslechu velmi podobný originálu, ale liší se ve spektru (viz obrázek 3.24). V důsledku došlo ke zkreslení výsledku agenta sítě GACELA, kdy hodnota SI-SDR jeho výstupního signálu vyšla záporná. Kvůli projekci ekvilibria došlo i k ostrým skokům v časovém průběhu jeho výstupu (viz obrázek 3.24), které při subjektivním poslechu způsobují výrazné lupnutí. S ohledem na délku mezery poskytla opět překvapivě dobrý výsledek metoda *gap-wise* Janssen. I tentokrát je pravděpodobnou příčinou vhodně zvolená poloha mezery ve stacionární části signálu.

Tab. 3.10: Experiment na nahrávce klavíru – změřené hodnoty objektivních metrik rekonstruovaného signálu pro mezeru délky 371 ms. Výsledky sítě GACELA a metody CE jsou zkráceny vlivem algoritmu PGHI.

Test klavír	Mask	Sparse-Analysis	GapJanssen	GACELA	CE
SDR (dB)	18,464	22,719	31,226	−26,628	22,163
MSL	0,520	0,210	0,053	0,425	0,161
Visqol	4,692	4,711	4,728	4,696	4,688



Obr. 3.25: Časové průběhy rekonstruovaného signálu metody CE a jednotlivých agentů v druhém testu sítě GACELA pro rockovou nahrávku.

Kvůli výsledkům prvního experimentu byla v druhém před spuštěním ekvilibria provedena projekce na výstupu sítě GACELA, aby byl zaručen původní kontext z nepoškozené části. Jeho výsledky znázorňuje tabulka 3.11. Pro vysokou paměťovou náročnost byla z ekvilibria vyřazena metoda *gap-wise* Janssen, která způsobovala vyčerpání paměti podobně jako inferenční funkce sítě CQTdiff+. Pro experiment byly stanoveny váhy $\mu_1 = 0,3$ pro agenta spektrální řídkosti a $\mu_2 = 0,7$ pro síť GACELA. Z časového průběhu metody řídkosti na obrázku 3.25 je patrná její neschopnost rekonstrukce pro mezeru délky 743 ms. Výstupní signál vykazuje velmi významný pokles energie v mezeře, který je pravděpodobnou příčinou obdobného jevu ve výsledku metody CE (viz obrázek 3.25). Výsledky objektivních metrik v tomto

testu jsou sporné. Přestože metoda řídkosti spektra dosáhla z hlediska subjektivního poslechu nejhoršího výsledku, získala nejvyšší hodnotu SDR. Její výsledek navíc nekoreluje s metrikami MSL a Visqol, které svědčí ve prospěch agenta sítě GACELA.

Změřená hodnota SDR v tomto experimentu nemusí být relevantní kvůli délce rekonstruované mezery. SDR porovnává po vzorcích podobnost dvou signálů. Šance, že rekonstruovaný signál bude pro rostoucí délku mezery odpovídat původnímu vzoru, je velmi malá, neboť v hudebím signálu klesá s delší mezerou šance na dodržení stacionarity. Výskyt nových událostí a změn v dynamice je pro dlouhé mezery pravděpodobnější, a proto se ani síť GACELA nesnaží přiblížit originálu. Jejím cílem je spíše zaplnit mezeru tak, aby výsledek podle subjektivního poslechu zněl dobře. Při současné schopnosti generovat nestacionární signál s dynamickými změnami dává tato síť velkou variabilitu v dosažení různorodých řešení, která se však od originálu mohou odchýlovat. Proto i přes aplikaci projekce dosáhla síť GACELA z hlediska metriky SI-SDR horšího výsledku než samotný maskovaný signál. Za relevantnější lze považovat výsledek metriky Visqol, který odpovídá i vjemu při subjektivním poslechu. Podle něj nejlepšího výsledku dosáhla síť GACELA samotná.

Z výsledku testování vyplývá, že metoda CE klade na agenty dodatečná omezení. Pokud je ekvilibrium použito s agenty, kteří sami dostatečně kvalitní řešení nalézt nedokážou, výsledek ekvilibria může být horší, než výsledek nejlepšího agenta pro tento případ. Test sítě GACELA demonstruje její omezení z hlediska použitých agentů a zachování kontextu v okolí mezery. Jedním z účelů tohoto testu bylo prozkoumat, zda ekvilibrium dokáže najít lepší řešení, než agenti i v případě, kdy jednotliví agenti řešení nalézt nedokážou. Výsledky testování hovoří v neprospěch této teze.

Tab. 3.11: Experiment na rockové nahrávce – změřené hodnoty objektivních metrik rekonstruovaného signálu pro mezeru délky 743 ms.

Test rock	Mask	Sparse-Analysis	GACELA	CE
SDR (dB)	10,052	10,074	7,634	9,932
MSL	0,882	0,484	0,211	0,328
Visqol	4,560	4,575	4,679	4,591

3.8.3 Test na různých vstupních signálech

Posledním testem, který proběhl v rámci této práce bylo porovnání průměrných výsledků získaných z několika různých vstupních signálů. Pro testování bylo vybráno 20 nahrávek klavíru s kmitočtem $f_{vz} = 22,05$ kHz vygenerovaných z MIDI souborů datasetu Maestro. Pro vygenerování zvuku piana byl použit plugin Mini Grand, který je součástí setu *First AIR Instruments Bundle*⁵ pro Pro Tools. Z metod byly pro agenty ekvilibria vybrány GACELA, *gap-wise* Janssen a syntetizující metoda spektrální řídkosti. Pro inferenci sítě GACELA byl použit model označený jako `real_data_240_32_240` natrénovaný na nahrávkách klavíru a definoval tak i délku mezery 371 ms. Velikost kroku metody CE byla zvýšena na hodnotu $\rho = 0,2$ a počet iterací ekvilibria byl nastaven na 20 pro každý testovací signál. Váhy agentů byly nastaveny následovně $\mu = [0,2; 0,3; 0,5]$. Nastavení agenta metody *gap-wise* Janssen zůstalo stejné jako ve všech předchozích testech (viz tabulka 3.3). Nastavení agenta syntetizující metody spektrální řídkosti bylo podobné jako u analyzující metody. Délka okna byla zvýšena na $N = 32768$ vzorků, délka skoku zůstala zachována na hodnotě $hop = 4096$ vzorků a velikost iteračního kroku byla zvýšena na hodnotu $\tau = 0,005$. Výsledky testování shrnuje tabulka 3.12, ve které jsou uloženy průměrné hodnoty všech metrik.

Tab. 3.12: Průměrné hodnoty objektivních metrik z testu 20 nahrávek klavíru. Signály obsahovaly jednu mezeru o délce 371 ms.

Test 20 nahrávek	Mask	Sparse-Synthesis	GapJanssen	GACELA	CE
$\overline{\text{SDR}}$ (dB)	12,208	14,513	15,052	9,756	14,860
$\overline{\text{MSL}}$	0,697	0,216	0,193	0,184	0,236
$\overline{\text{Visqol}}$	4,471	4,569	4,614	4,577	4,547

Z výsledků vyplývá, že metoda CE poskytla druhé nejlepší řešení podle SDR. Metriky MSL a Visqol jej však vyhodnotily jako nejhorší. Pro definitivní rozhodnutí o kvalitě rekonstrukce byly výsledné nahrávky podrobeny i subjektivnímu poslechu. Z něj bylo kvalitu rekonstrukce velmi obtížné posoudit, neboť výslednou rekonstrukci významně ovlivnila poloha mezery. Řešení, která byla nejlepší z pohledu objektivních metrik nebyla z hlediska subjektivního poslechu vždy uspokojivá. Výsledek tohoto testu je sporný. Pro přesnější a objektivnější výsledky by bylo vhodné použít model sítě GACELA, který by zvládal zaplnit i kratší mezery do délky 200 ms. Natrénovaný model s těmito parametry však není k dispozici.

⁵<https://www.avid.com/plugins/first-air-instrument-bundle>

3.9 Hodnocení

Z výsledků testování je patrné, že na kvalitní rekonstrukci signálu navržené metody CE má vliv mnoho parametrů. Velikost kroku ρ a počet iterací ekvilibria má vliv na rychlost nalezení řešení (sekce 3.4). Počet zvolených iterací je složitější, neboť v závislosti na volbě vah jednotlivých agentů, konverguje metoda CE v krátkodobém horizontu (malém počtu iterací) k jinému řešení, než v dlouhodobém (sekce 3.6). Vliv volby počátečních podmínek na nalezené řešení je v dlouhodobém horizontu zanedbatelný (sekce 3.7). Zásadní vliv má naopak délka mezery a volba jednotlivých agentů ekvilibria. Závislost na délce mezery definují sami použití agenti. V průběhu testování se ukázalo, že obě metody řídkosti samy dokážou bez výrazného poklesu energie doplnit pouze mezery ve stacionární oblasti signálu do délky 150 ms. V první části testování se pro mezery do 300 ms pokles energie projevoval pouze v agentu řídkosti samotném, ale nijak zásadně v ekvilibriu. V testu popsaném v sekci 3.8.2 pro mezeru délky 743, ms se již výsledek metody řídkosti v ekvilibriu projevil a pravděpodobně zde vedl k poklesu energie v mezeře. Metoda *gap-wise* Janssen dominovala ve všech testech z hlediska objektivních metrik. Nevýhodou je její velká výpočetní náročnost, neboť jedna iterace této metody v ekvilibriu zabrala nejvíce času ze všech agentů. Vysoká je i její paměťová náročnost a pro mezery delší, než 371 ms skončila vyčerpáním operační paměti.

U sítě GACELA se velmi problematickou ukázala být rekonstrukce fáze, neboť je aplikována i na nepoškozený kontext. Rekonstruované fázové spektrum se liší od fázového spektra originálního signálu a v částech kontextu dochází při součtu signálu k poklesu amplitudy. Pokud je na výstup sítě GACELA aplikována projekce, kontext zůstane zachován, ale v mezeře ve výstupu sítě samotné vznikne fázový skok, který způsobí slyšitelné lupnutí na začátku a konci mezery. V ekvilibriu se však tento fázový skok ve většině případů ztratil. Nejlepších výsledků dosáhla metoda CE pro mezery od 50 do 300 ms při použití pouze klasických metod inpaintingu (viz test 3.8.1).

Z testu 3.8.3 na 20 signálech s mezerou délky 371 ms a testu 3.8.2 na mezeře délky 743 ms vyplývá, že metoda CE není pravděpodobně schopná zaplnit při inpaintingu ve vstupním signálu dlouhé mezery, pokud toho nejsou schopni všichni jednotliví agenti. V průběhu testování bylo v těchto případech vždy lepší řešení jednoho z agentů samostatně. Skripty pro spuštění všech testů provedených v rámci této práce jsou součástí přílohy B.

Závěr

Tato práce zkoumá problematiku doplňování chybějících úseků audiosignálů pojmenovanou též jako audio inpainting. Zaměřuje se na metodu *consensus equilibrium* (CE), která umožňuje kombinovat klasické a moderní metody inpaintingu jako takzvané agenty. Ze zmíněných metod používá v různých kombinacích dvě metody spektrální řídkosti, jednu metodu autoregrese a 2 metody založené na hlubokém učení.

Metoda byla otestována na vliv parametrů a různá nastavení jejích jednotlivých agentů pro nahrávky klavíru, vážné hudby a rockové hudby s mezerami v rozsahu od 50 do 743 ms. Zkoumán byl vliv parametrů na nalezení konkrétního řešení, rychlost konvergence k tomuto řešení a kvalitu rekonstrukce nahrávky z hlediska objektivních metrik. Ukázalo se, že výstup metody CE implementované v této práci je velmi závislý na volbě a nastavení jejích agentů. Pokud nějaký z nich zavádí omezení na vstupní signál, pak se tato omezení mohou nepříznivě projevit i v ekvilibriu a jejich nedodržení často vede k horšímu výsledku ve prospěch rekonstrukce pomocí některého z agentů samostatně. Pokud vstupní signál tato omezení splňoval, překonala metoda CE ve většině případů samostatná řešení svých agentů. Nejeфекtivnější se ukázala být v případech, kdy všichni agenti samostatně dokázali poskytnout dobré řešení.

Metoda *gap-wise* Janssen, která byla součástí většiny testů ekvilibria, měla výrazný pozitivní vliv na jeho výsledky v první části testování. Metody spektrální řídkosti naopak dokázaly řešení ekvilibria pro dlouhé mezery významně zhoršit. Pro získání přesnějších informací o chování ekvilibria by bylo vhodné otestovat v iteracích i vliv změny hyperparametrů jeho agentů, neboť ten byl z časových důvodů testován pouze odděleně na agentech samotných a následně zafixován nebo byl převzat z literatury. Přínosné by mohlo být i otestování progresivně se měnících vah agentů. Výchozím bodem by byla jejich rovnost a na základě metrik odhadnutých z poškozeného a restaurovaného signálu by byly váhy agentů v každé iteraci upraveny ve prospěch nejschopnějšího z nich. Přínosné by bylo i zavedení ukončovací podmínky, neboť metoda CE dokáže být v závislosti na použitých agentech velmi časově náročná.

Ačkoliv metoda CE nedospěla vždy k nejlepšímu řešení, ve většině případů našla druhé nejlepší nebo alespoň ne nejhorší řešení. V situacích, kdy některý z agentů nebyl schopný mezeru zaplnit, ale většina ostatních poskytla dle objektivních metrik řešení dobré, směřovala k dobrému řešení i metoda CE. Lze na ni proto pohlížet i jako na pojistku, která dokáže zabránit nalezení špatného řešení. Stačí pouze, aby většina jejích agentů byla samostatně schopna mezeru zaplnit. Díky tomu zvyšuje z hlediska objektivních metrik schopnost dobré rekonstrukce oproti jednotlivým metodám svých agentů.

Literatura

- [1] JANSSEN, Augustus; VELDHUIS, Raymond a VRIES, Lodewijk. „Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes“. Online. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. April 1986, roč. 34, č. 2, s. 317-330. Dostupné z: <https://doi.org/10.1109/TASSP.1986.1164824>. [cit. 2024-11-15].
- [2] MOKRÝ, Ondřej a RAJMÍČ, Pavel. „Audio Inpainting: Revisited and Reweighted“. Online. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. 2020, roč. 28, s. 2906-2918. Dostupné z: <https://doi.org/10.1109/TASLP.2020.3030486>. [cit. 2024-12-08].
- [3] MOLINER, Eloi a VÄLIMÄKI, Vesa. „Diffusion-Based Audio Inpainting“. Online. May 2023. Dostupné z: <https://doi.org/10.48550/arXiv.2305.15266>. [cit. 2024-12-08].
- [4] ETTER, Walter. „Restoration of a discrete-time signal segment by interpolation based on the left-sided and right-sided autoregressive parameters“. Online. *IEEE Transactions on Signal Processing*. 1996, roč. 44, č. 5, s. 1124-1135. Dostupné z: <https://doi.org/10.1109/78.502326>. [cit. 2024-11-15].
- [5] LAGRANGE, Mathieu; MARCHAND, Sylvain a RAULT, Jean-Bernard. „Long Interpolation of Audio Signals Using Linear Prediction in Sinusoidal Modeling“. Online. *Journal of the Audio Engineering Society*. October 2005, roč. 53, s. 891-905. Dostupné z: https://www.researchgate.net/publication/228372207_Long_Interpolation_of_Audio_Signals_Using_Linear_Prediction_in_Sinusoidal_Modeling. [cit. 2024-12-08].
- [6] PERRAUDIN, Nathanaël; HOLIGHAUS, Nicki; MAJDAK, Piotr a BALAZS, Peter. „Inpainting of Long Audio Segments With Similarity Graphs“. Online. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. 2018, roč. 26, č. 6, s. 1083-1094. Dostupné z: <https://doi.org/10.1109/TASLP.2018.2809864>. [cit. 2024-12-08].
- [7] MARAFIOTI, Andrés; HOLIGHAUS, Nicki; MAJDAK, Piotr a PERRAUDIN, Nahanaël. „Audio inpainting of music by means of neural networks“. Online. 2022, s. 1-13. Dostupné z: <https://arxiv.org/abs/1810.12138>. [cit. 2024-11-29].

- [8] ADLER, Amir; EMIYA, Valentin; JAFARI, Maria G.; GRIBONVAL, Rémi a PLUMBLEY, Mark D. „Audio Inpainting“. Online. *IEEE Transactions on Audio, Speech, and Language Processing*. 2012, roč. 20, č. 3, s. 922-932. Dostupné z: <https://doi.org/10.1109/TASL.2011.2168211>. [cit. 2024-12-08].
- [9] EBNER, Pirmin Philipp a ELTELTH, Amr. „Audio inpainting with generative adversarial network.“ Online. 2020, s. 1-11. Dostupné z: <https://arxiv.org/abs/2003.07704>. [cit. 2024-11-30].
- [10] MARAFIOTI, Andrés; HOLIGHAUS, Nicki a PERRAUDIN, Nathanaël. „GACELA: A Generative Adversarial Context Encoder for Long Audio Inpainting of Music“. Online. *IEEE Journal of Selected Topics in Signal Processing*. 2021, roč. 15, č. 1, s. 120-131. Dostupné z: <https://doi.org/10.1109/JSTSP.2020.3037506>. [cit. 2024-12-08].
- [11] PRŮŠA, Zdeněk a SØNDERGAARD, Peter L. „Real-time spectrogram inversion using phase gradient heap integration“. Online. *Proceedings of the 19th International Conference on Digital Audio Effects (DAFx-16)*. September 2016, s. 17-21. Dostupné z: <https://www.dafx.de/paper-archive/details/KyhYU44R13wwkkqNs3bgPw>. [cit. 2024-12-10].
- [12] MIRZA, Mehdi a OSINDERO, Simon. „Conditional Generative Adversarial Nets“. Online. 2014, s. 1-6. Dostupné z: <https://arxiv.org/abs/1411.1784>. [cit. 2024-12-02].
- [13] MOLINER, Eloi; LEHTINEN, Jaakko a VÄLIMÄKI, Vesa. „Solving Audio Inverse Problems with a Diffusion Model“. Online. 2023. Dostupné z: <https://arxiv.org/abs/2210.15228>. [cit. 2024-12-04].
- [14] BROWN, Judith a PUCKETTE, Miller. „An efficient algorithm for the calculation of a constant Q transform“. Online. *Journal of the Acoustical Society of America*. 1992, roč. 1992, č. 92, s. 2698-2701. Dostupné z: <https://doi.org/10.1121/1.404385>. [cit. 2024-12-05].
- [15] MIOTELLO, Federico; PEZZOLI, Mirco; COMANDUCCI, Luca; ANTONACCI, Fabio a SARTI, Augusto. „Deep Prior-Based Audio Inpainting Using Multi-Resolution Harmonic Convolutional Neural Networks“. Online. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. 2024, roč. 32, s. 113-123. Dostupné z: <https://doi.org/10.1109/TASLP.2023.3324556>. [cit. 2024-12-11].

- [16] ELAD, Michael; MILANFAR, Peyman a RUBISTEIN, Ron. „Analysis versus synthesis in signal priors“. Online. *14th European Signal Processing Conference*. Roč. 2006, s. 1-5. Dostupné z: <https://ieeexplore.ieee.org/document/7071162>. [cit. 2024-12-07].
- [17] L. COMBETTES, Patrick a PESQUET, Jean-Christophe. „Proximal Splitting Methods in Signal Processing“. Online. 2010. Dostupné z: <https://arxiv.org/abs/0912.3522>. [cit. 2024-12-07].
- [18] MOKRÝ, Ondřej. „Moderní metody restaurace poškozených audiosignálů“. Online, Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská Kolejni 4, 612 00 Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav matematiky, 2019. Dostupné z: https://www.vut.cz/studenti/zav-prace?%20zp_id=113107. [cit. 2024-12-08].
- [19] BOYD, Stephen; PARIKH, Neal; CHU, Eric; PELEATO, Borja a ECKSTEIN, Jonathan. „Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers“. Online. *Foundations and Trends in Machine Learning*. 2011, roč. 3, s. 1-122. Dostupné z: <https://doi.org/10.1561/22000000016>. [cit. 2025-05-12].
- [20] BUZZARD, Gregory T.; CHAN, Stanley H.; SREEHARI, Suhas a BOUMAN, Charles A. „Plug-and-Play Unplugged: Optimization Free Reconstruction using Consensus Equilibrium“. Online. 2018. Dostupné z: <https://arxiv.org/abs/1705.08983>. [cit. 2025-02-21].
- [21] TANAKA, Tomoro; YATABE, Kohei a OIKAWA, Yasuhiro. „UPGLADE: Unplugged Plug-and-Play Audio Declipper Based on Consensus Equilibrium of DNN and Sparse Optimization“. Online. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2023, s. 1-5. Dostupné z: <https://doi.org/10.1109/ICASSP49357.2023.10095928>. [cit. 2025-03-13].
- [22] BAUSCHKE, Heinz a COMBETTES, Patrick. „Convexity and Nonexpansiveness“. In: *Convex Analysis and Monotone Operator Theory in Hilbert Space*. 2. Springer, 2017, s. 59-60. ISBN 978-1-4419-9466-0.
- [23] MOKRÝ, Ondřej a RAJMIC, Pavel. „The best autoregressive approach to audio inpainting is gap-wise Janssen“. Online. 2025. Dostupné z: <https://arxiv.org/abs/2403.04433>. [cit. 2025-05-13].

- [24] ROUX, Jonathan Le; WISDOM, Scott; ERDOGAN, Hakan a HERSHEY, John R. „SDR – Half-baked or Well Done?“ Online. In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing*. Piscataway, New Jersey, USA: Institute of Electrical and Electronics Engineers (IEEE), 2019, s. 626-630. ISBN 9781479981328. Dostupné z: <https://doi.org/10.1109/ICASSP.2019.8683855>. [cit. 2025-05-17].
- [25] TORCOLI, Matteo; KASTNER, Thorsten a HERRE, Jürgen. „Objective Measures of Perceptual Audio Quality Reviewed: An Evaluation of Their Application Domain Dependence“. Online. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. 2021, roč. 29, s. 1530-1541. Dostupné z: <https://doi.org/10.1109/TASLP.2021.3069302>. [cit. 2025-05-17].
- [26] YAMAMOTTO, Ryuichi; SONG, Eunwoo a KIM, Jae-Min. „Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram“. Online. 2020. Dostupné z: <https://arxiv.org/abs/1910.11480>. [cit. 2025-05-17].
- [27] CHINEN, Michael; LIM, Felicia S. C.; SKOGLUND, Jan; GUREEV, Nikita; O’GORMAN, Feargus et al. „ViSQOL v3: An Open Source Production Ready Objective Speech and Audio Metric“. Online. 2020. Dostupné z: <https://arxiv.org/abs/2004.09584>. [cit. 2025-05-17].

Seznam symbolů a zkratek

AR	Autoregrese – Autoregression
CE	Consensus Equilibrium
cGAN	Podmíněná generativní adversariální síť – conditional Generative Adversarial Network
CQT	Konstantní Q transformace – Constant Q Transform
DNN	Hluboká neuronová síť – Deep Neural Network
f_{vz}	Vzorkovací kmitočet
GAN	Generativní adversariální síť – Generative Adversarial Network
ICQT	Inverzní konstantní Q transformace – Inverse Constant Q Transform
ISTFT	Inverzní krátkodobá Fourierova transformace – Inverse Short Time Fourier Transform
MIDI	Digitální rozhraní hudebního nástroje – Musical Instrument Digital Interface
MSL	Multi-resolution Spectral Loss
PGHI	Postupná integrace gradientu fáze – Phase Gradient Heap Integration
RMS	Efektivní hodnota signálu – Root Mean Square
SDR	Signal-to-Distortion Ratio
SI-SDR	Scale-invariant Signal-to-Distortion Ratio
STFT	Krátkodobá Fourierova transformace – Short Time Fourier Transform
Visqol	Virtual Speech Quality Objective Listener
WGAN	Wassersteinova generativní adversariální síť – Wasserstein Generative Adversarial Network

Seznam příloh

A	Prostředí a programovací jazyk	95
B	Obsah elektronické přílohy	97

A Prostředí a programovací jazyk

V počátcích této práce byly implementovány a otestovány 2 metody inpaintingu založené na řídkosti spektra pro platformu Matlab. Prostředí bylo zvoleno díky uživatelské přívětivosti při ladění a možnosti rychle vyvinout prototyp. Jelikož jsou však neuronové sítě napsány v jazyce Python, byl v neprospěch Matlabu zvolen jako výchozí prostředí pro metodu CE a metody řídkosti byly do Pythonu přepsány. Pro správu balíčků byla využita kombinace virtuálního prostředí Anaconda a systému pro správu pip.

Velmi problematickou se ukázala již instalace sítí kvůli jejich technické kompatibilitě se současnými verzemi Pythonu, a to především u sítě GACELA, neboť její poslední aktualizace proběhla před pěti lety v červnu roku 2020. V popisku sítě není explicitně uvedeno, na jaké verzi Pythonu byla testována. Při pokusu nainstalovat ji pomocí Pythonu verze 3.12, který je v době psaní této práce poslední aktuální verzí, docházelo k četným selháním kvůli závislosti na externí knihovně `ltfatpy`¹, kterou GACELA používá pro časově-frekvenční analýzu a syntézu signálu. Tato knihovna je portem originální knihovny `ltfat` vyvinuté pro Matlab. Stejně jako GACELA nebyla řadu let aktualizována a při četných pokusech o její manuální instalaci skrze správce balíčků `pip` docházelo k řadě potíží. Části knihovny `ltfatpy` jsou napsány v jazyce Cython, který překládá zdrojový kód do jazyka C. Zde vzniká konflikt při pokusu kompilovat kód s příliš aktuální verzí Pythonu, neboť automaticky generovaný kód obsahuje konstrukce s ukazateli, které jsou v novějších verzích Cythonu označeny jako nepodporované a kompilace knihovny díky tomu končí chybou. Tento kód je automaticky generován a mazán při kompilaci, takže jeho manuální oprava není možná. Z hlediska kompatibility jsou problematické i další závislosti této knihovny na balíčcích FFTW3 a LAPACK napsaných v jazyce C, které `ltfatpy` vyžaduje pro spuštění instalace. Jedinou možností bylo downgradovat verzi Pythonu a najít takovou kombinaci knihoven, která půjde pro tuto downgradovanou verzi nainstalovat. GACELA disponuje ještě jednou problematickou závislostí a tou je knihovna `essentia`², neboť verze, kterou GACELA požaduje v souboru `requirements.txt` je pro Python 3.12 nekompatibilní. Postupným snižováním verze Pythonu a knihoven byla nalezena kombinace kompatibilní pro spuštění. Výslednou verzí pro instalaci sítě GACELA byl Python 3.7.16, ve kterém se přes značné potíže podařilo zprovoznit obě kritické knihovny. Soubor `requirements.txt` pro instalaci GACELY byl několikrát upraven a výsledné verze závislostí jsou součástí přílohy B této práce.

¹<https://pypi.org/project/ltfatpy/>

²<https://pypi.org/project/essentia/>

Sít CQTdiff ve své dokumentaci udává požadavek na Python verze alespoň 3.8, který je s požadavky sítě GACELA v rozporu. CQTdiff+ je nástupcem sítě CQTdiff, a jelikož se jedná o novější verzi, lze předpokládat, že požadavek na verzi Pythonu je také 3.8. Tato situace má dvě možná řešení:

1. Společné virtuální prostředí, ve kterém se nachází všechny sítě, kódy agentů i kód pro ekvilibrium.
2. Oddělená virtuální prostředí, kdy se každá síť spouští s vlastní verzí Pythonu a ekvilibrium tyto sítě volá skrze bash a subprocess.

První možnost sice přinesla komplikace, neboť bylo potřeba síť CQTdiff nainstalovat do nepodporované verze Pythonu, ale byla nakonec vybrána jako výchozí hlavně kvůli možnostem při ladění kódu, které spouštění sítě skrze bash velmi omezuje. Závislosti knihovny CQTdiff definované v souboru `requirements.txt` byly proto po vzoru sítě GACELA upraveny tak, aby bylo možné síť spustit se starším Pythonem.

Knihovna `ltfatpy` má ještě jedno omezení a tím je absence podpory pro Windows, čímž i použití sítě GACELA omezuje na MacOS a Linux. Pro práci na operačním systému Windows bylo proto nutné použít Linuxový subsystém pro Windows (WSL) a omezit tak stejným způsobem podporu metody CE v případě, že síť GACELA použije jako jednoho z agentů. Kvůli výše zmíněným skutečnostem **nelze** navrženou metodu CE pro inpainting spustit na operačním systému MS Windows.

Pro instalaci metody CE a všech jejích závislostí včetně jednotlivých agentů je potřeba nainstalovat všechny knihovny obsažené v souboru `requirements-all.txt` v kořenovém adresáři přílohy B. Pro zprovonění sítí GACELA, CQTdiff a CQTdiff+ je nutné stáhnout natrénované modely. K tomu slouží předpřipravené skripty jejich autorů:

GACELA/`download_checkpoints.py` pro síť GACELA

CQTdiff/`download_weights_and_examples.sh` pro síť CQTdiff

CQTdiff+/`notebooks/demo_inpainting_spectrogram.ipynb` pro síť CQTdiff+

Důležité je i zdůraznit, že veškerý kód metody CE byl testován pouze na zařízení nepodporující rozhraní CUDA. Z časových důvodů nebylo možné otestovat metodu i na jiném zařízení, a proto nelze garanovat plnou kompatibilitu pro zařízení CUDA využívající.

B Obsah elektronické přílohy

Elektronická příloha obsahuje všechny kódy v jazicích Python a Matlab realizované v rámci této práce a další grafy z výsledků testování, které nebyly přidány přímo do praktické části. Přílohy byly i kódy neuronových sítí pro inpainting. Ve struktuře přílohy jsou zmíněny pouze autorské skripty vytvořené v rámci této práce nebo skripty pro stažení natrénovaných modelů. Soubory jsou strukturovány následovně:

```
/.....kořenový adresář příloženého archivu
├── AR.....zdrojové kódy pro agenty autoregrese
│   ├── initAR.py.....inicializace AR agentů
│   ├── processARex.py.....definice agenta extrapolace
│   ├── processGapJanssen.py.....definice agenta gap-wise Janssen
│   ├── runARex.py.....samostatné otestování metody extrapolace
│   └── runJanssen.py.....samostatné otestování metody gap-wise Janssen
├── audio.....testovací zvukové soubory
├── CQTdiff.....zdrojové kódy sítě CQTdiff
│   ├── download_weights_and_examples.sh.....skript pro stažení modelů
│   ├── initCQTdiff.py.....inicializace agenta CQTdiff
│   ├── processCQTdiff.py.....definice agenta CQTdiff
│   └── runCQTdiff.py.....samostatné otestování sítě CQTdiff
├── CQTdiff+.....zdrojové kódy sítě CQTdiff+
│   └── runCQTplus.py.....samostatné otestování sítě CQTdiff+
├── GACELA.....zdrojové kódy sítě GACELA
│   ├── download_checkpoints.py.....skript pro stažení natrénovaných modelů
│   ├── initGacela.py.....inicializace sítě GACELA
│   ├── processGacela.py.....definice agenta GACELA
│   └── runGacela.py.....samostatné otestování sítě GACELA
├── plot.....další průběhy z výsledků testování metod řídkosti
├── SparseInpainting.....zdrojové kódy pro agenty řídkosti
│   ├── MatlabPrototype.....kódy prototypu metod řídkosti v Matlabu
│   ├── analysisMethod.py.....definice metody analýzy
│   ├── initSparse.py.....inicializace agentů řídkosti spektra
│   ├── operators.py.....definice operátorů
│   ├── processSparse.py.....definice agenta řídkosti
│   ├── runSparse.py.....samostatné otestování metod řídkosti spektra
│   └── synthesisMethod.py.....definice metody syntézy
├── consensusEquilibrium.py.....definice tříd ekvilibria a agentů
├── gapGenerator.py.....pomocné metody pro generování mezer
├── globalUtils.py.....globální metody používané napříč projektem
├── metrics.py.....definice objektivních metrik
├── plotter.py.....metody pro vykreslování grafů
├── requirements-all.txt.....seznam všech potřebných balíčků a závislostí
├── runCE.py.....hlavní spouštěcí skript pro testy DP
├── test_metrics.py.....skript pro otestování metrik
└── testCEinpainting.py.....spouštěcí skript pro demonstraci inpaintingu
```